

(19) RÉPUBLIQUE FRANÇAISE  
INSTITUT NATIONAL  
DE LA PROPRIÉTÉ INDUSTRIELLE  
PARIS

(11) N° de publication :  
à n'utiliser que pour les  
commandes de reproduction

**2 597 142**

(21) N° d'enregistrement national :

**87 04971**

(51) Int Cl<sup>a</sup> : E 05 B 49/00, 19/16.

(12) **DEMANDE DE BREVET D'INVENTION**

A1

(22) Date de dépôt : 8 avril 1987.

(30) Priorité : US. 8 avril 1986, n° 849 472.

(43) Date de la mise à disposition du public de la  
demande : BOP « Brevets » n° 42 du 16 octobre 1987.

(80) Références à d'autres documents nationaux appa-  
rantes :

(71) Demandeur(s) : Société dite : SCHLAGE LOCK COM-  
PANY. — US.

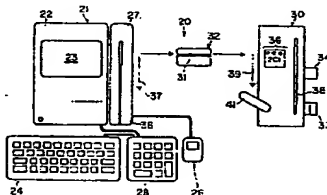
(72) Inventeur(s) : Victor H. Vee, Thomas W. Crosley, Ronald  
D. Lichty, Wayne Davison, John R. Goldberg, Leonard L.  
Hofheins, Charles A. Vollum et Stephen H. Vollum.

(73) Titulaire(s) :

(74) Mandataire(s) : Cabinet Beau de Loménie.

(54) Système de serrure électronique cryptographique et procédé de fonctionnement.

(57) La présente invention concerne un système de serrure adapté à fonctionner sur la base du codage et de la vérification d'un message de données porté par un milieu d'enregistrement discret telle qu'une carte magnétique 32, comprenant un premier moyen d'ordinateur adapté à appliquer une clef cryptographique privée pour coder le message de données; des moyens 27 pour écrire le message de données codées sur le milieu; des moyens de serrure 30 comprenant un verrou 33, cette serrure agissant en réponse à la vérification du message de données codées pour ouvrir le verrou; et un second moyen d'ordinateur dans la serrure pour appliquer une clef de cryptographie publique au message de données codées pour vérifier le message de données.



FR 2 597 142 - A1

BEST AVAILABLE COPY

1/8

<u>Clef</u>	<u>Serrure</u>		<u>Serrure</u>												
			se recombine?	S'ouvre?											
Premier	Avant	Après													
<table><tr><td>N<sub>1</sub></td></tr></table>	N <sub>1</sub>	<table><tr><td>Premier</td><td>Second</td></tr><tr><td>N<sub>1</sub></td><td>N<sub>2</sub></td></tr></table>	Premier	Second	N <sub>1</sub>	N <sub>2</sub>	<table><tr><td>Premier</td><td>Second</td></tr><tr><td>N<sub>1</sub></td><td>N<sub>2</sub></td></tr></table>	Premier	Second	N <sub>1</sub>	N <sub>2</sub>	<table><tr><td>non</td></tr></table>	non	<table><tr><td>oui</td></tr></table>	oui
N <sub>1</sub>															
Premier	Second														
N <sub>1</sub>	N <sub>2</sub>														
Premier	Second														
N <sub>1</sub>	N <sub>2</sub>														
non															
oui															
Second															
<table><tr><td>N<sub>2</sub></td></tr></table>	N <sub>2</sub>	<table><tr><td>Premier</td><td>Second</td></tr><tr><td>N<sub>1</sub></td><td>N<sub>2</sub></td></tr></table>	Premier	Second	N <sub>1</sub>	N <sub>2</sub>	<table><tr><td>Premier</td><td>Second</td></tr><tr><td>N<sub>2</sub></td><td>N<sub>3</sub></td></tr></table>	Premier	Second	N <sub>2</sub>	N <sub>3</sub>	<table><tr><td>oui</td></tr></table>	oui	<table><tr><td>oui</td></tr></table>	oui
N <sub>2</sub>															
Premier	Second														
N <sub>1</sub>	N <sub>2</sub>														
Premier	Second														
N <sub>2</sub>	N <sub>3</sub>														
oui															
oui															
Troisième-non utilisé															
<table><tr><td>N<sub>3</sub></td></tr></table>	N <sub>3</sub>	<table><tr><td>Premier</td><td>Second</td></tr><tr><td>N<sub>2</sub></td><td>N<sub>3</sub></td></tr></table>	Premier	Second	N <sub>2</sub>	N <sub>3</sub>	<table><tr><td>Premier</td><td>Second</td></tr><tr><td>N<sub>2</sub></td><td>N<sub>3</sub></td></tr></table>	Premier	Second	N <sub>2</sub>	N <sub>3</sub>	<table><tr><td>Non utilisé</td></tr></table>	Non utilisé	<table><tr><td></td></tr></table>	
N <sub>3</sub>															
Premier	Second														
N <sub>2</sub>	N <sub>3</sub>														
Premier	Second														
N <sub>2</sub>	N <sub>3</sub>														
Non utilisé															
Quatrième															
<table><tr><td>N<sub>4</sub></td></tr></table>	N <sub>4</sub>	<table><tr><td>Premier</td><td>Second</td></tr><tr><td>N<sub>2</sub></td><td>N<sub>3</sub></td></tr></table>	Premier	Second	N <sub>2</sub>	N <sub>3</sub>	<table><tr><td>Premier</td><td>Second</td></tr><tr><td>N<sub>2</sub></td><td>N<sub>3</sub></td></tr></table>	Premier	Second	N <sub>2</sub>	N <sub>3</sub>	<table><tr><td>non</td></tr></table>	non	<table><tr><td>non</td></tr></table>	non
N <sub>4</sub>															
Premier	Second														
N <sub>2</sub>	N <sub>3</sub>														
Premier	Second														
N <sub>2</sub>	N <sub>3</sub>														
non															
non															

FIG. 1

<u>Clef</u>	<u>Serrure</u>		<u>Serrure</u>									
	Avant	Après	Se recombine?	S'ouvre								
Premier												
<table><tr><td>Premier</td><td>Second</td></tr><tr><td>N<sub>1</sub></td><td>N<sub>2</sub></td></tr></table>	Premier	Second	N <sub>1</sub>	N <sub>2</sub>	<table><tr><td>N<sub>1</sub></td></tr></table>	N <sub>1</sub>	<table><tr><td>N<sub>2</sub></td></tr></table>	N <sub>2</sub>	<table><tr><td>oui</td></tr></table>	oui	<table><tr><td>oui</td></tr></table>	oui
Premier	Second											
N <sub>1</sub>	N <sub>2</sub>											
N <sub>1</sub>												
N <sub>2</sub>												
oui												
oui												
Second												
<table><tr><td>Premier</td><td>Second</td></tr><tr><td>N<sub>2</sub></td><td>N<sub>3</sub></td></tr></table>	Premier	Second	N <sub>2</sub>	N <sub>3</sub>	<table><tr><td>N<sub>2</sub></td></tr></table>	N <sub>2</sub>	<table><tr><td>N<sub>3</sub></td></tr></table>	N <sub>3</sub>	<table><tr><td>oui</td></tr></table>	oui	<table><tr><td>oui</td></tr></table>	oui
Premier	Second											
N <sub>2</sub>	N <sub>3</sub>											
N <sub>2</sub>												
N <sub>3</sub>												
oui												
oui												
Troisième-non utilisé												
<table><tr><td>Premier</td><td>Second</td></tr><tr><td>N<sub>3</sub></td><td>N<sub>4</sub></td></tr></table>	Premier	Second	N <sub>3</sub>	N <sub>4</sub>	<table><tr><td>N<sub>2</sub></td></tr></table>	N <sub>2</sub>	<table><tr><td>N<sub>3</sub></td></tr></table>	N <sub>3</sub>	<table><tr><td>Non utilisé</td></tr></table>	Non utilisé	<table><tr><td></td></tr></table>	
Premier	Second											
N <sub>3</sub>	N <sub>4</sub>											
N <sub>2</sub>												
N <sub>3</sub>												
Non utilisé												
Quatrième												
<table><tr><td>Premier</td><td>Second</td></tr><tr><td>N<sub>4</sub></td><td>N<sub>5</sub></td></tr></table>	Premier	Second	N <sub>4</sub>	N <sub>5</sub>	<table><tr><td>N<sub>2</sub></td></tr></table>	N <sub>2</sub>	<table><tr><td>N<sub>3</sub></td></tr></table>	N <sub>3</sub>	<table><tr><td>non</td></tr></table>	non	<table><tr><td>non</td></tr></table>	non
Premier	Second											
N <sub>4</sub>	N <sub>5</sub>											
N <sub>2</sub>												
N <sub>3</sub>												
non												
non												

FIG. 2

<u>Clef</u>	<u>Serrure</u>		<u>Serrure</u>															
	Avant	Après	se recombine?	S'ouvre?														
Premier																		
<table><tr><td>Premier</td><td>Second</td></tr><tr><td>N<sub>1</sub></td><td>N<sub>2</sub></td></tr></table>	Premier	Second	N <sub>1</sub>	N <sub>2</sub>	<table><tr><td>Premier</td><td>Second</td></tr><tr><td>N<sub>1</sub></td><td>N<sub>2</sub></td></tr></table>	Premier	Second	N <sub>1</sub>	N <sub>2</sub>	<table><tr><td>Premier</td><td>Second</td></tr><tr><td>N<sub>1</sub></td><td>N<sub>2</sub></td></tr></table>	Premier	Second	N <sub>1</sub>	N <sub>2</sub>	<table><tr><td>non</td></tr></table>	non	<table><tr><td>oui</td></tr></table>	oui
Premier	Second																	
N <sub>1</sub>	N <sub>2</sub>																	
Premier	Second																	
N <sub>1</sub>	N <sub>2</sub>																	
Premier	Second																	
N <sub>1</sub>	N <sub>2</sub>																	
non																		
oui																		
Second																		
<table><tr><td>Premier</td><td>Second</td></tr><tr><td>N<sub>2</sub></td><td>N<sub>3</sub></td></tr></table>	Premier	Second	N <sub>2</sub>	N <sub>3</sub>	<table><tr><td>Premier</td><td>Second</td></tr><tr><td>N<sub>1</sub></td><td>N<sub>2</sub></td></tr></table>	Premier	Second	N <sub>1</sub>	N <sub>2</sub>	<table><tr><td>Premier</td><td>Second</td></tr><tr><td>N<sub>2</sub></td><td>N<sub>3</sub></td></tr></table>	Premier	Second	N <sub>2</sub>	N <sub>3</sub>	<table><tr><td>oui</td></tr></table>	oui	<table><tr><td>oui</td></tr></table>	oui
Premier	Second																	
N <sub>2</sub>	N <sub>3</sub>																	
Premier	Second																	
N <sub>1</sub>	N <sub>2</sub>																	
Premier	Second																	
N <sub>2</sub>	N <sub>3</sub>																	
oui																		
oui																		
Troisième-non utilisée																		
<table><tr><td>Premier</td><td>Second</td></tr><tr><td>N<sub>3</sub></td><td>N<sub>4</sub></td></tr></table>	Premier	Second	N <sub>3</sub>	N <sub>4</sub>	<table><tr><td>Premier</td><td>Second</td></tr><tr><td>N<sub>2</sub></td><td>N<sub>3</sub></td></tr></table>	Premier	Second	N <sub>2</sub>	N <sub>3</sub>	<table><tr><td>Premier</td><td>Second</td></tr><tr><td>N<sub>2</sub></td><td>N<sub>3</sub></td></tr></table>	Premier	Second	N <sub>2</sub>	N <sub>3</sub>	<table><tr><td>Non utilisée</td></tr></table>	Non utilisée		
Premier	Second																	
N <sub>3</sub>	N <sub>4</sub>																	
Premier	Second																	
N <sub>2</sub>	N <sub>3</sub>																	
Premier	Second																	
N <sub>2</sub>	N <sub>3</sub>																	
Non utilisée																		
Quatrième																		
<table><tr><td>Premier</td><td>Second</td></tr><tr><td>N<sub>4</sub></td><td>N<sub>5</sub></td></tr></table>	Premier	Second	N <sub>4</sub>	N <sub>5</sub>	<table><tr><td>Premier</td><td>Second</td></tr><tr><td>N<sub>2</sub></td><td>N<sub>3</sub></td></tr></table>	Premier	Second	N <sub>2</sub>	N <sub>3</sub>	<table><tr><td>Premier</td><td>Second</td></tr><tr><td>N<sub>2</sub></td><td>N<sub>3</sub></td></tr></table>	Premier	Second	N <sub>2</sub>	N <sub>3</sub>	<table><tr><td>non</td></tr></table>	non	<table><tr><td>non</td></tr></table>	non
Premier	Second																	
N <sub>4</sub>	N <sub>5</sub>																	
Premier	Second																	
N <sub>2</sub>	N <sub>3</sub>																	
Premier	Second																	
N <sub>2</sub>	N <sub>3</sub>																	
non																		
non																		

FIG. 3

2/8

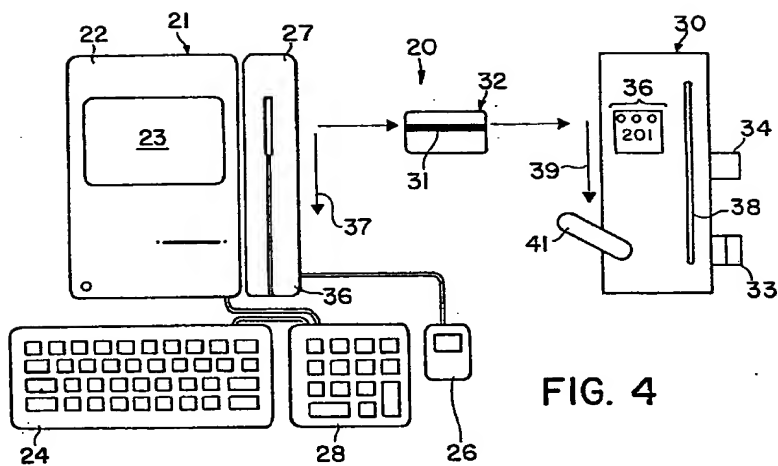


FIG. 4

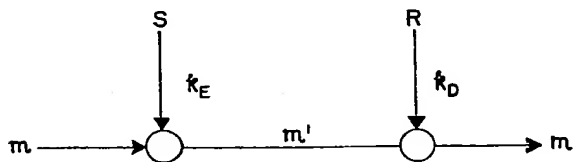


FIG. 5

FIG. 6

3/8



4/8

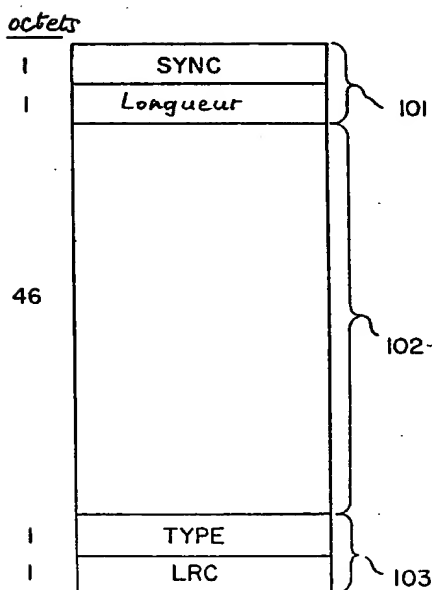
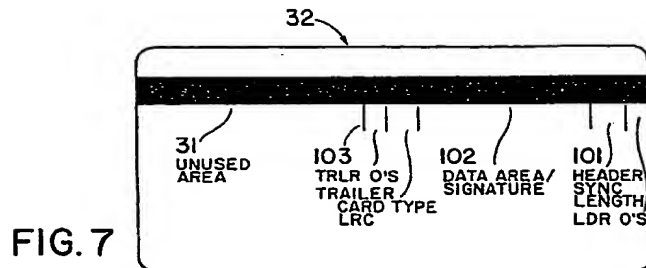


FIG. 8

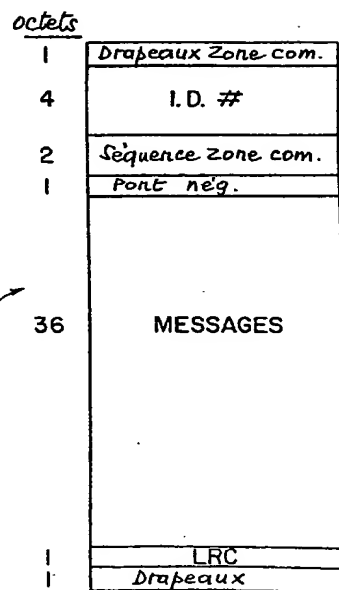


FIG. 9

5/8

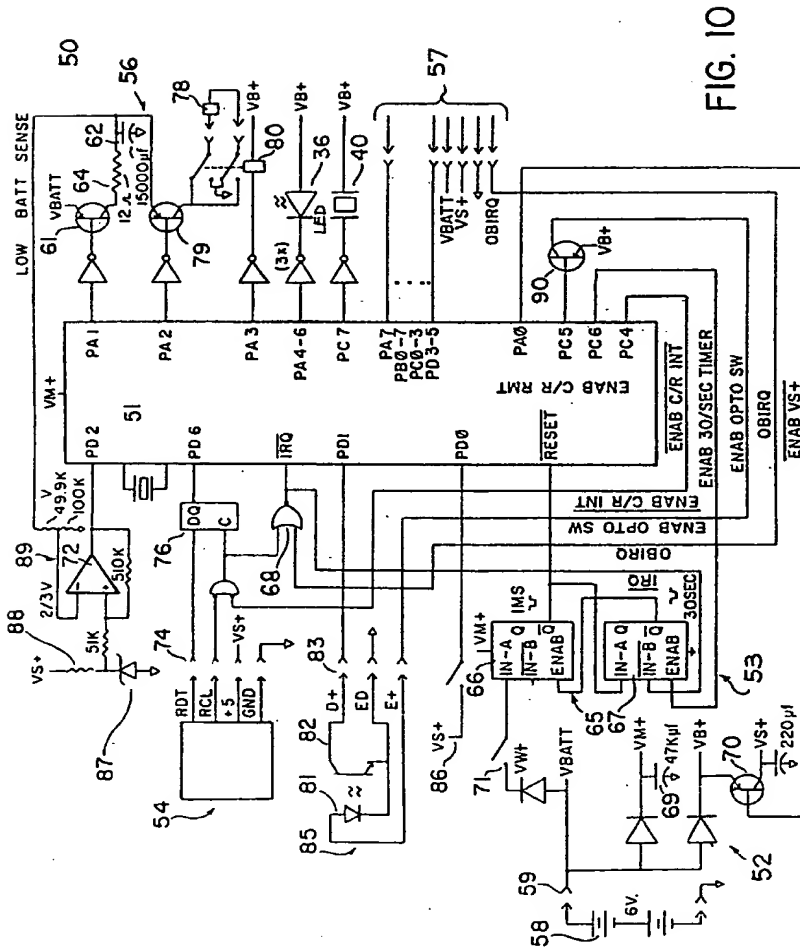


FIG. 10

6/8

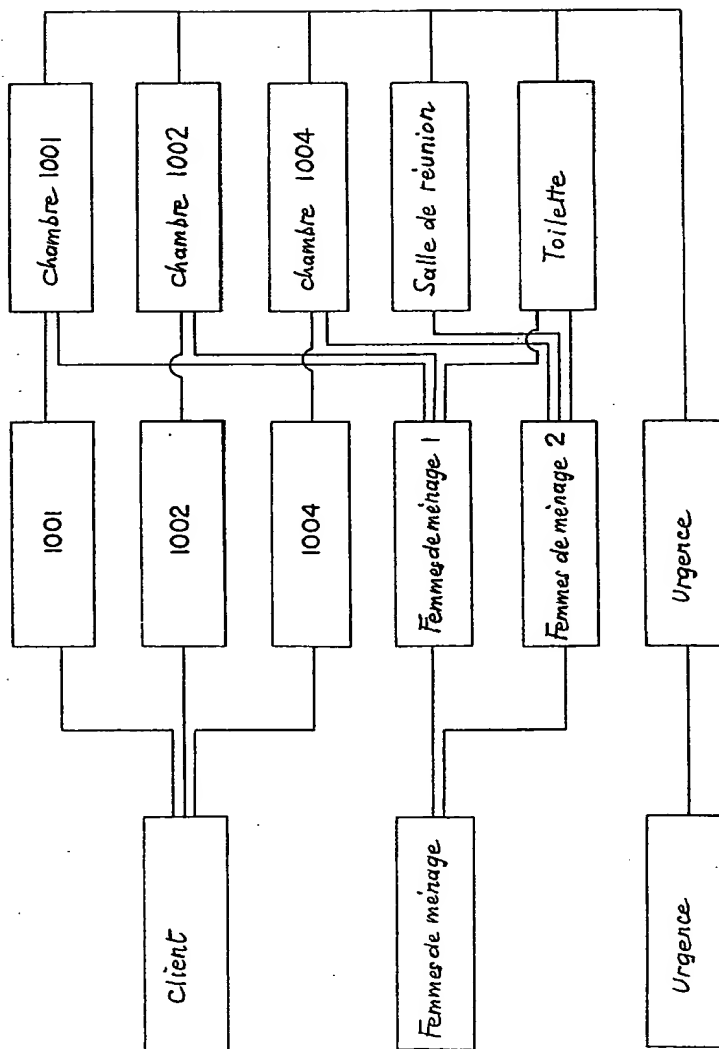


FIG. 12

7/8

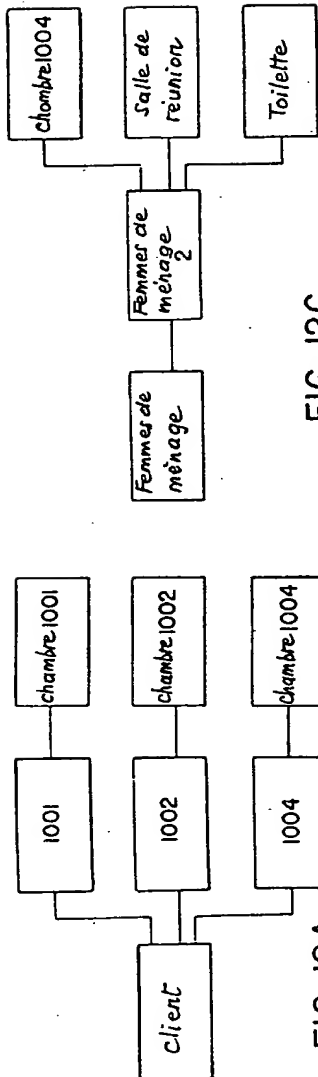


FIG. 12C

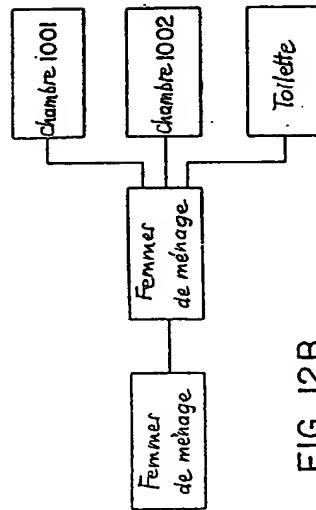
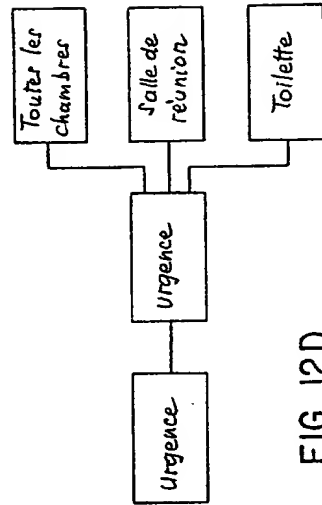
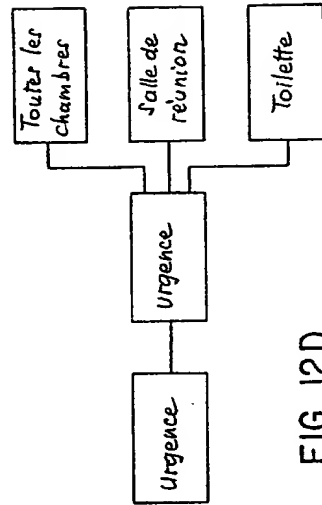


FIG. 12D



8/8

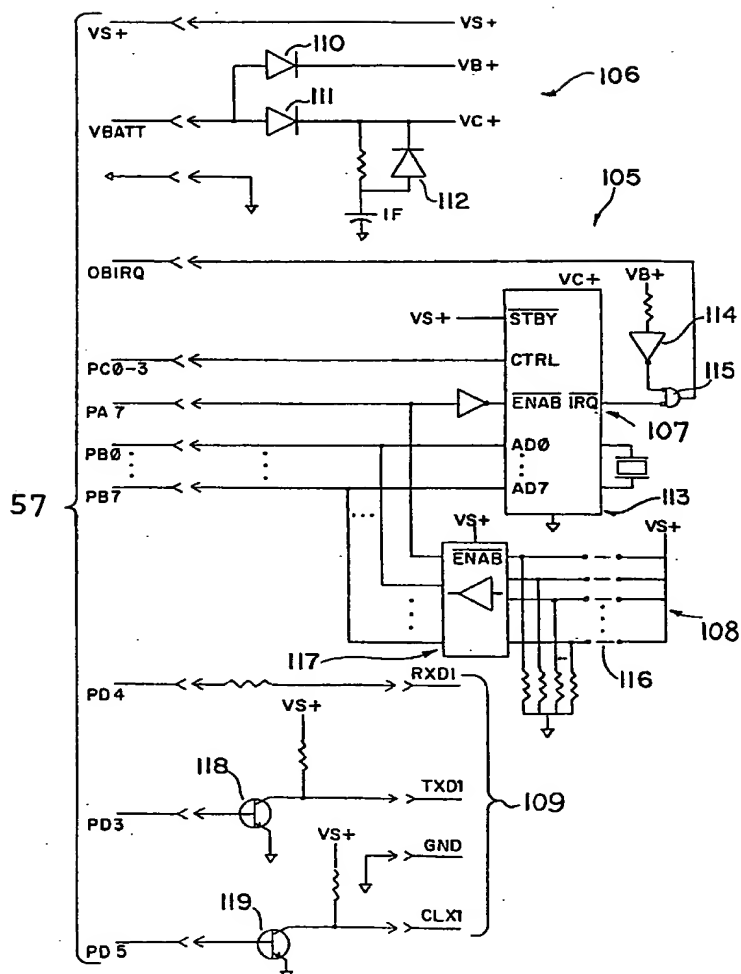


FIG. 13

SYSTEME DE SERRURE ELECTRONIQUE CRYPTOGRAPHIQUE  
ET PROCEDE DE FONCTIONNEMENT

La présente invention concerne des serrures électroniques et des systèmes de verrouillage électronique, des systèmes de verrouillage électronique qui utilisent des cartes servant de clefs codées à distance et, en particulier, un système de verrouillage électronique qui utilise une cryptographie à clef publique.

Le procédé de fonctionnement d'une serrure électronique et de mise à jour des informations de programme dans cette serrure sur la base des informations codées dans une carte servant de clef (carte-clef), c'est-à-dire sans communication directe avec l'ordinateur utilisé pour coder la carte-clef, est limité par plusieurs facteurs. Ceux-ci comprennent la relativement faible quantité de mémorisation de données qui est disponible sur la carte-clef et dans la serrure électronique elle-même, et les capacités limitées de vitesse et de calcul des microprocesseurs qui sont utilisés dans de telles serrures. Ces limitations d'espace et de calcul sont très importantes quand on considère que la carte-clef doit contenir un certain type de code ou de combinaison d'identification secret, ainsi que des instructions pour actionner une ou plusieurs serrures choisies (ou empêcher leur actionnement) et en ce que la serrure doit à la fois valider la carte et mettre en oeuvre les instructions.

A ce jour, seulement quelques systèmes viables possibles sont disponibles et utilisent une carte-clef programmée à distance pour commander le fonctionnement mécanique et la programmation d'une serrure électronique. Ces approches antérieures sont notamment décrites dans les brevets des Etats-Unis d'Amérique No 3 800 284 de Zucker, 3 860 911 de Hinman, 3 821 704 et RE 29 259 de Sabsay, et 4 511 946 de McGahan.

Dans le système décrit dans le brevet de Zucker, à tout instant donné avant la reprogrammation par une nouvelle clef, la serrure contient deux types d'informations de codes :

premièrement, le numéro de code précédent et deuxièmement le numéro de code séquentiel suivant. La clef est codée par une combinaison unique. Ce système est conçu de sorte que, de façon supposée, quand une nouvelle clef valide convenablement séquencée est fournie, la combinaison de clef s'adapte à la combinaison séquentielle suivante de la serrure et amène la serrure à s'ouvrir et à se reprogrammer. Pendant la reprogrammation, un générateur de fonction dans la serrure utilise la combinaison précédemment mémorisée dans la serrure pour produire une combinaison en cours et la combinaison séquentielle suivante. Lors d'un usage suivant de cette même clef, la serrure s'ouvrira parce que le premier code de serrure est égal au code de clef en cours. Toutefois, la serrure n'est pas recombinaison ou reprogrammée à ce moment car la nouvelle combinaison séquentielle a déjà été re-séquencée et n'est plus égale au code de clef. Après recombinaison par la nouvelle clef, le code de serrure en cours n'est plus égal au code de la clef immédiatement précédente et, en conséquence, cette clef n'ouvrira plus la serrure.

Le système de Hinman utilise deux combinaisons, à la fois dans la serrure et dans la clef, mais il fonctionne de façon similaire à celle utilisée par Zucker.

Le verrou électronique décrit dans le brevet de Sabsay est l'inverse de celui utilisé par Zucker en ce qu'on attribue à la serrure une seule combinaison alors que la clef comprend deux champs ou combinaisons. Les champs de la clef sont : un premier champ ou numéro d'autorisation qui est le code précédemment autorisé, et un second champ ou numéro de clef qui contient le code autorisé en cours. Quand une clef est présentée à la serrure, si le second champ ou champ "en cours" égale le numéro de la serrure unique, la serrure s'ouvre. Si le code "précédent" dans le premier champ d'autorisation est égal au numéro de serrure, la serrure se recombine et s'ouvre. Quand une nouvelle clef est présentée à la serrure, le code précédent dans le premier champ de la clef doit être égal au numéro de serrure en cours de sorte que la serrure se recombine et s'ouvre. Après quoi, chaque fois que cette clef est utilisée (avant recombinaison par la clef suivante)

le numéro de serrure mis à jour sera égal au code en cours dans le second champ de la clef et la serrure s'ouvrira mais ne se recombina pas.

5 Le brevet de McGahan utilise des première et seconde combinaisons dans la serrure ainsi que dans la clef. Les combinaisons de la serrure et de la clef sont séquentielles en ce que la seconde combinaison est le numéro séquentiel suivant au-dessus de la première combinaison. Pendant une utilisation, si la première combinaison de clef est égale à la première combinaison de serrure  
10 et si la seconde combinaison de clef est égale à la seconde combinaison de serrure, la serrure s'ouvre. Si cette égalité n'existe pas, mais que la première combinaison de clef est égale à la seconde combinaison de serrure, la serrure s'ouvre et se recombine. Ainsi, quand la clef suivante convenablement séquencée  
15 est présentée à la serrure, la première combinaison de clef sera égale à la seconde combinaison de serrure et la serrure s'ouvrira et se recombina. Ensuite, jusqu'à ce qu'une nouvelle clef recombine la serrure, les première et seconde combinaisons de serrure et de clef sont égales et la clef en cours ouvrira la  
20 serrure mais ne l'amènera pas à se recombina. Les clefs antérieures ne pourront pas ouvrir ou recombina la serrure car ni l'une ni l'autre des deux égalités requises n'existent entre les codes de serrure et de clef.

Toutefois, à la connaissance des demandeurs, aucun des  
25 systèmes de serrure électronique présentement disponibles, y compris le système de McGahan n'élimine le problème de la mise en séquence qui prend place quand la séquence de clefs et la séquence de serrure se décalent, par exemple parce que l'on n'utilise pas une carte dûment fournie et séquencée. Cette situation est  
30 représentée dans les figures 1 à 3 pour Zucker, Sabsay et McGahan, respectivement. Dans chaque cas, des première et seconde clefs validement fournies et séquencées sont utilisées de façon prévue et recombina la serrure comme cela est programmé. Toutefois, la troisième clef qui est également validement fournie et séquencée  
35 n'est pas utilisée. Ceci peut survenir simplement parce qu'un client n'entre pas dans sa chambre ou n'utilise pas une chambre particulière dans une suite de chambres.

Quelle que soit la raison, après que l'on a manqué d'utiliser la troisième carte-clef dûment fournie, la quatrième carte et la suivante n'actionneront pas la serrure.

En outre, dans le système de serrure électronique existant, la fonction de sécurité et les fonctions d'actionnement sont en compétition en raison de l'espace limité disponible dans la carte-clef et la serrure, d'où il résulte que l'une ou l'autre ou les deux fonctions peuvent être limitées à un degré indésirable ou inacceptable. Par exemple, on souhaite disposer d'une large sélection d'utilisation de serrures possibles, telle que des niveaux clients, des niveaux de suite, des zones communes, etc.; et pouvoir accéder à ces différentes combinaisons de serrure ou de niveaux de serrures au moyen d'une carte-clef unique. A ce jour, les limitations physiques inhérentes des cartes-clefs et des serrures électroniques ont limité même les systèmes de verrouillages électroniques les plus souples à un choix unique au niveau d'une serrure quelconque entre huit ou neuf niveaux maîtres possibles et commande par une quelconque carte individuelle d'un niveau ou serrure maître unique.

Au vu de ce que précède, un objet de la présente invention est de prévoir un système de verrouillage électronique et un procédé d'actionnement du système dans lequel la sécurité est prévue par une cryptographie à clef publique.

Un autre objet associé est de prévoir un tel système de verrouillage électronique et son procédé de fabrication dans lequel la fonction de sécurité est séparée des messages portés sur la carte-clef codant le champ de message en utilisant une cryptographie numérique du type signature.

Un autre objet associé de la présente invention est de prévoir un système de verrouillage électronique et son procédé d'actionnement dans lequel une carte-clef communique avec la serrure électronique au moyen d'un protocole souple accroissant ainsi le nombre d'opérations qui peuvent être réalisées au niveau de serrures individuelles et commandées ou effectuées par des clefs individuelles.

Dans un premier mode de réalisation, la présente

invention implique le procédé de chiffage du champ de message d'une carte-clef en utilisant une cryptographie à clef publique, puis le déchiffage du message de carte codée au niveau de la serrure pour valider le message avant sa mise en oeuvre.

5 Dans un mode de réalisation actuellement préféré, le système et le procédé de verrouillage électronique selon l'invention utilisent un nombre  $x$  et une fonction modulo;  $x^2 \bmod n = m$ , où  $n$  est la clef publique et  $m$  est le message. Le message codé ou signé  $x$  est transmis par l'intermédiaire de la carte-clef à la  
10 serrure qui déchiffre ou décrypte le message de carte sous-jacent  $m$  à partir du message chiffré  $x$  en calculant  $x^2 \bmod n$ .

Dans un mode de réalisation spécifique conçu pour faciliter le calcul de  $x$ , une clef privée est utilisée qui comprend une paire de nombres premiers  $p$  et  $q$  qui sont déterminés  
15 de sorte que  $m = pq$ . La clef publique est déterminée de sorte qu'elle comprend seulement deux facteurs : les clefs privées  $p$  et  $q$ . Le message chiffré  $x$  est calculé à partir du message  $m$  en calculant  $x \bmod n$ . Ce calcul peut être réalisé en une durée raisonnable en utilisant les clefs privées  $p$  et  $q$ .

20 L'utilisation ci-dessus d'une cryptographie à clef publique permet l'utilisation d'un protocole de communication souple qui assure lui-même de nombreux avantages décrits ci-après.

En outre, l'invention comprend diverses fonctions spécifiques de circuit électronique et de serrure mécanique  
25 décrites ci-après.

Ces objets, caractéristiques et avantages de la présente invention seront décrits plus en détail ci-après en relation avec les dessins joints parmi lesquels :

30 Les figures 1 à 3 représentent trois approches classiques de validation de clefs et de recombinaison et d'ouverture de serrures en réponse et décrit le problème de séquençement qui résulte couramment du fait qu'une clef valide n'est pas utilisée ;

35 La figure 4 est une représentation schématique du système de verrouillage électronique d'ensemble selon la présente invention ;

La figure 5 représente schématiquement l'approche de cryptographie à clef publique qui est incorporée dans le système de verrouillage électronique selon l'invention et utilisée dans son fonctionnement ;

5 La figure 6 représente le programme de multiplicité réitérative pour réduire la mémoire de la serrure et le calcul dans la serrure requis pour élever au carré le message codé x ;

Les figures 7, 8 et 9, respectivement, décrivent une carte magnétique indiquée à titre d'exemple, l'organisation des  
10 informations hexadécimales sur la carte, et l'organisation de la zone de données ;

La figure 10 représente schématiquement le circuit de commande utilisé dans la serrure électronique ;

15 La figure 11 représente schématiquement une organisation de niveau de serrure ;

Les figures 12 et 12A à 12D représentent à titre d'exemple les relations entre les niveaux principaux, les zones, et l'ouverture de serrure ; et

20 La figure 13 est une représentation schématique du circuit à option renforcée.

#### A. SYSTEME D'ENSEMBLE

Un mode de réalisation actuellement préféré 20 d'un système de serrure électronique incorporant la présente invention est représenté en figure 4. Le système de serrure électronique  
25 comprend une console codeuse 21, qui comprend un ordinateur 22 et un afficheur 23, un clavier 24, un module de commande dit à souris 26, et un module de lecture/écriture de cartes 27. La console peut comprendre un clavier 28 pour faciliter l'introduction de données numériques dans la mémoire de l'ordinateur.

30 Le système de serrure électronique 20 comprend également une serrure électronique autonome 30 contenant un microprocesseur qui est programmé par des informations codées sur la bande magnétique 31 de cartes 32 pour effectuer sélectivement des opérations de verrouillage et de déverrouillage d'un pêne 33 et  
35 d'un verrou 34. Des voyants vert, jaune et rouge, typiquement des diodes photoluminescentes (LED) désignées collectivement par la

référence 36 indiquent l'état de la serrure 30. Egalement, un vibreur audible 40 (figure 10) est incorporé dans la serrure. Il faut noter que la carte (ou autre milieu) et les modules de lecture et d'écriture peuvent être de tout type connu, tels que magnétique, optique, ou infrarouge. En ce qui concerne le système de verrouillage de façon générale, l'homme de l'art pourra mettre en oeuvre simplement le système de verrouillage en utilisant d'autres composants sur la base de la description fournie ici.

Dans un mode de réalisation particulier, la console utilise un système d'ordinateur de la société dite Apple connu sous la marque MacIntosh et un module commercialement disponible de lecture/écriture de cartes. De même, la serrure électronique utilise un microprocesseur 6805 et un module lecteur de cartes classique. En outre, une mémoire à disques d'ordinateur sera typiquement prévue dans le module de console. Pour des opérations importantes, on peut souhaiter connecter plusieurs consoles et des mémoires à disques durs associées utilisant un réseau local.

En fonctionnement, les données de la carte 32 sont introduites dans la console 21 en utilisant le clavier 24, la souris 26 et/ou le clavier 28 et les données sont chiffrées par l'ordinateur 21. La carte 32 est alors amenée à passer dans la fente 36 du module de lecture/écriture de cartes 27 comme cela est indiqué par la flèche 37, pour enregistrer les données chiffrées sur la carte. Au niveau de la serrure 30, la carte-clef magnétique 32 est amenée à passer dans la fente 38, comme cela est indiqué par la flèche 39 pour fermer un commutateur de veille 71 (figure 10) et activer ainsi le microprocesseur 51, et également pour permettre au module lecteur de carte de la serrure de retrouver les données codées. Le microprocesseur déchiffre ou décrypte alors les données et détermine si le message codé x est un message valide m. Si le message de données est valide, il est utilisé pour programmer le verrou et/ou pour actionner la serrure. Par exemple, et comme cela est décrit plus complètement ci-après, des données transmises par une carte-clef valide convenablement séquencée 32 déterminent le degré de sécurité fourni par le pêne 33 et le verrou 34 et quand, et si, la poignée 41 sera capable de

déverrouiller la serrure. En outre, les informations communiquées par la carte-clef 32 à la serrure 30 comprennent diverses formes pour le verrou, telles que des instructions pour qu'il s'ouvre quand la poignée 41 est tournée ; pour qu'il s'ouvre seulement si le verrou 34 n'est pas mis ; pour déverrouiller une femme de chambre ; etc...

Le système 20 assure une sécurité en codant le message de la carte-clef en utilisant un chiffage spécifique de signature numérique et une méthodologie de déchiffage qui est rapidement exécuté au niveau de la console et de la serrure. L'incorporation d'un protocole flexible assure une plus grande flexibilité de fonctionnement que ce qui est disponible dans les systèmes de verrouillage électronique antérieurs. En outre, un programme de séquençement est utilisé et n'est pas soumis au problème de saut d'étape exposé précédemment. Ces caractéristiques ainsi que d'autres sont exposées ci-après.

#### B. SIGNATURE NUMERIQUE

Comme cela a été mentionné, le système de verrou électronique selon l'invention est adapté à utiliser une forme modifiée de cryptographie à clef publique à signature numérique, en dépit des limitations de mémorisation de données et de calculs qui sont inhérentes à un tel système. Comme cela est représenté en figure 5, de façon générale quand on utilise une cryptographie à clef publique, un émetteur, S, chiffre un message m en utilisant une clef de chiffage  $k_E$  et transmet ou transfère le message de texte chiffré codé m' vers le récepteur R. Le récepteur utilise une clef de déchiffage  $k_D$  pour transférer en retour le message codé en le message en clair original m.

Le mode général de cryptographie ci-dessus peut-être mis en oeuvre selon deux modes particuliers différents : la cryptographie classique et la cryptographie publique. En cryptographie classique, les clefs de chiffage et de déchiffage sont identiques :  $k_E = k_D = k$ . Cette approche inclut la norme de cryptage numérique classique bien connue DES. Un problème crucial avec les systèmes de cryptage classiques, si ceux-ci sont appliqués à des systèmes de verrouillage électronique et qu'il

serait nécessaire de communiquer la clef commune  $k$  à l'émetteur et au récepteur. La sécurité de cette clef deviendrait alors cruciale pour la sécurité du système lui-même. Par exemple, la sécurité de la clef peut être mise en brèche par une analyse technique ou une inspection de la serrure, ou par une rupture de confidentialité de la part de l'une quelconque des nombreuses personnes qui doivent nécessairement avoir accès à la clef.

En cryptographie publique,  $k_D$  est différent de  $k_E$ . La cryptographie de type publique recouvre deux sous-espèces ou options. D'abord, la clef de chiffrement  $k_E$  peut être publique et la clef de déchiffrement  $k_D$  secrète auquel cas tout le monde peut envoyer un message mais seul le récepteur  $R$  peut le décoder. Un exemple de cette approche réside dans le système de courrier électronique.

La seconde approche de cryptographie à clef publique consiste en l'inverse de la première. Ainsi, la clef de chiffrement  $k_E$  est maintenue secrète et la clef de déchiffrement  $k_D$  est publique. Par suite, seul l'émetteur  $S$  qui connaît la clef secrète  $k_D$  peut transmettre un message codé valide mais tout le monde peut déchiffrer le message codé pour vérifier que le message codé est valide. C'est l'approche dite à signature numérique et ceci est préférable pour des considérations de sécurité. Une application à titre d'exemple du système est décrite dans l'ouvrage de Meyer et Matyas intitulé "Cryptography" publié chez John Wiley and Sons en 1982 en particulier dans la partie du chapitre 2 intitulée Block Cyphers concernant les algorithmes RSA en pages 33 à 48. Cet ouvrage sera considéré ici comme connu.

L'algorithme RSA (nommé d'après ses inventeurs) implique fondamentalement l'évaluation d'une fonction modulo du type  $x^k \bmod n = m$  où  $x$  est un message qui, quand il est élevé à la puissance de la clef  $k$  et divisé par un nombre composite  $n$  fournit un reste  $m$ .

La signature numérique de la clef de verrouillage électronique selon l'invention est une version modifiée du type d'algorithme RSA ayant la forme  $x^k \bmod n = m$ . L'utilisation de cette fonction modulo pour transmettre les messages codés implique

de calculer au niveau de la console une racine carrée  $x$  telle que  $x^2 \bmod n = m$ , c'est-à-dire que  $x^2$  divisé par  $n$  fournit le reste  $m$ . Le quotient n'est pas utilisé. Ici,  $m$  est le message à transmettre,  $n$  est la clef publique et  $x$  est le message codé  $m'$  de la figure 5.

Au niveau de la serrure, la fonction  $x^2 \bmod n$  est calculée pour retrouver ou déchiffrer le message codé  $m$ .

La sécurité apportée par la présente application de cryptographie à clef publique pour des systèmes de verrouillage est directement proportionnelle à la dimension du nombre de la clef publique. Ainsi, assurer une sécurité qui, en pratique, ne peut être violée implique l'utilisation d'une très grande clef publique. La présente version du système de verrouillage 20 utilise une clef publique  $n$  d'environ 111 chiffres. A partir de la théorie des nombres des résidus quadratiques, on peut prouver que trouver les racines carrées d'un nombre composite est aussi difficile que de factoriser ce nombre. Ainsi, en choisissant la clef publique à 111 chiffres ( $n$ ) comme étant le produit de deux grands entiers, ce problème de factorisation peut être rendu très difficile. Factoriser un grand nombre peut nécessiter des mois ou même des années même pour l'ordinateur sophistiqué le plus rapide tel que le super ordinateur CRAY 2, et encore plus pour l'ordinateur capable mais plus lent et moins sophistiqué de la console et le système d'ordinateur beaucoup plus lent et à plus faible capacité utilisé dans la serrure 30. En outre, à la connaissance des demandeurs, le conflit existant entre les grands nombres qui sont requis pour la sécurité et la très grande vitesse de fonctionnement ( $\leq 0,5$  secondes) qui est requise pour un actionnement commode de la serrure peut seulement être résolu en utilisant les séquences de codage/décodage suivantes qui ont été prévues.

L'algorithme de codage/décodage recouvre trois groupes fondamentaux d'étapes : un précalcul de diverses valeurs qui sont indépendantes de la valeur du message ; un codage et un chiffrement du message de la carte-clef au niveau de la console ; et une vérification et un rétablissement du message de la carte-clef au niveau de la serrure (ou de la console). Ces trois algorithmes

partagent un ensemble de variables globales communes :

1.  $p, q$  : une paire de nombres premiers connus  
seulement de la console qui constitue la clef secrète ;

2.  $n$  : la clef publique, le produit de  $p$  et  $q$ , ses  
seuls facteurs ;

3.  $p14, q14$  : les exposants utilisés pour trouver des  
racines partielles ;

4.  $p2, q2$  : les racines partielles de 2 ; et

5.  $kp, kq$  : les coefficients de combinaison - ceux-ci  
sont utilisés pour combiner deux racines partielles. Les trois  
étapes sont décrites ci-dessous.

#### 1. Précacul

Cet algorithme calcule les valeurs nécessaires dans le  
processus de chiffage. Il est exécuté une fois à chaque  
initialisation de la console. Son but est de réduire la durée de  
chiffage d'un message en précalculant les valeurs qui sont  
indépendantes de la valeur du message.

En utilisant les nombres premiers choisis,  $p$  et  $q$  cet  
algorithme calcule la clef publique ( $n$ ), les exposants ( $p14$  et  
 $q14$ ), les racines carrées partielles ( $p2$  et  $q2$ ), et les  
coefficients de combinaison ( $kp$  et  $kq$ ). Ces valeurs sont  
mémorisées dans des variables globales présentées ci-dessus.

L'algorithme pour précalculer  $n, p14, q14, p2, q2, kp,$   
 $kq$  en utilisant  $p$  et  $q$  implique les étapes suivantes :

#### Etapes

#### Explications

1a.  $p = le P$

1b.  $q = le Q$

2.  $n = p * q$

3.  $p14 = (p+1) \text{ div } 4$

4.  $q14 = (q+1) \text{ div } 4$

Sauvegarder les nombres

premiers  $p$  et  $q$  de la clef  
secrète.

Calculer la valeur de clef pu-  
blique  $n$  en multipliant  $p$  et  $q$ .  
Calculer l'exposant de racine  
partielle de  $p$  en ajoutant  
1 et en divisant par 4.

Calculer l'exposant de racine  
partielle  $q$ -ième de la même  
façon.

12

5.  $p2 = \text{puissance}(2, p14, p)$  Trouver  $p2$  de sorte que  
 $p2 \cdot \text{mod } p = \pm 2.$
6.  $q2 = \text{puissance}(2, q14, q)$  Trouver  $q2$  de sorte que  
 $q2 \cdot q2 \text{ mod } q = \pm 2.$
- 5 7.  $kp = q \cdot \text{puissance}(q, p-2, p)$  Trouver  $kp$  tel que  $kp \text{ mod } q = 0$ ,  
et  $kp \text{ mod } p = 1.$
8.  $kq = p \cdot \text{puissance}(p, q-2, q)$  Trouver  $kq$  de sorte que  
 $kq \text{ mod } q = 1$  et  $kq \text{ mod } p = 0.$

## 2. Message de signature

10 Comme cela a été mentionné, signer un message  $m$  consiste à trouver une valeur  $x$  telle  $x^2 \text{ mod } n = m$ . Seulement 25 % des valeurs possibles de  $m$  ont de telles racines. En requérant que  $m \text{ mod } 4 = 2$ , des réglages peuvent être effectués pendant le processus de signature et de vérification pour permettre la

15 signature de toute valeur de message légale.

L'algorithme de signature (chiffage) calcule d'abord des racines partielles de  $m$  par rapport à  $p$  et  $q$ , synchronise ensuite les racines partielles en doublant  $m$  si nécessaire. Enfin, les deux racines partielles sont combinées pour former la racine

20 par rapport à  $n$ .

Les étapes de l'algorithme de signature sont :

	<u>Etapes</u>	<u>Explications</u>
	1. $mp = m \text{ mod } p$	$mp$ est le résidu de $m \text{ mod } p$ .
25	2. $mq = m \text{ mod } q$	$mq$ est le résidu de $m \text{ mod } q$ .
	3. $xp = \text{puissance}(mp, p14, p)$	trouver $xp$ tel que $xp \cdot xp \text{ mod } p = \pm mp$ .
	4. $xq = \text{puissance}(mq, q14, q)$	Trouver $xq$ tel que $xq \cdot xq \text{ mod } q = \pm mq$ .
30	5. $tp = xp \cdot xp \text{ mod } p$	Calculer $xp \cdot xp \text{ mod } p$ .
	6. $tq = xq \cdot xq \text{ mod } q$	Calculer $xq \cdot xq \text{ mod } q$ .
	7. SI $(mp = tp) \neq (mq = tq)$	Si les signes relatifs diffèrent,
	COMMENCER ALORS	ce doit être la signature $2m$ ,
	$xp := xp \cdot p2 \text{ mod } p$	en conséquence trouver $xp$ tel
35	$xq := xq \cdot q2 \text{ mod } q$	que $xp \cdot xp \text{ mod } p = \pm 2 \cdot m \text{ mod } p$
		et $xq$ tel que $xq \cdot xq \text{ mod } q = \pm 2 \cdot m \text{ mod } q$ .



Il existe un second problème critique impliquant la mise en oeuvre de la cryptographie à signature numérique pour la technologie de serrures électroniques qui implique l'ordinateur de serrure. Alors que le microordinateur 6805 couramment utilisé dans le verrou 30 est relativement rapide et fournit une relativement grande quantité de mémoire vive (192 octets) et de mémoire morte (4096 octets), un tel microprocesseur de l'art antérieur fournit encore une très petite mémoire et une faible capacité de calcul en comparaison des exigences de calcul d'un très grand nombre tel que  $x' \bmod n$ . En outre, la mémoire vive (RAM) de travail disponible est encore réduite à environ 100 octets puisqu'environ 50 octets sont requis pour d'autres fonctions de verrouillage électronique. Plus simplement, il n'y a pas suffisamment de mémoire de travail RAM pour préserver un nombre codé  $x$  d'environ 46 octets et en même temps développer son produit  $x^2$  de longueur binaire double comme on devrait normalement le faire.

Ces limitations deviennent encore plus importantes quand on les considère à la lumière des besoins en conflits précédemment mentionnés de rendre maximum la dimension du nombre calculé  $x$  pour augmenter la sécurité et en même temps de satisfaire l'exigence que les calculs soient effectués en moins de 0,5 secondes pour empêcher un retard inacceptable après passage de la carte dans la fente de lecture de la serrure 38. En bref, et en plus du rendement de calcul qui est requis au niveau de la console et est assuré par l'algorithme de factorisation de  $p, q$  décrit ci-dessus, une grande efficacité de calcul est également requise pour calculer  $x^2 \bmod n$  très rapidement au niveau de la serrure avec la mémoire de travail RAM sévèrement limitée.

La présente invention inclut un mode de calcul qui assure l'efficacité souhaitée. Cet algorithme permet le calcul de  $x^2$  dans la même mémoire de travail RAM requise pour mémoriser  $x$ . L'algorithme est décrit ci-après en relation avec le processus d'élévation au carré du nombre à quatre chiffres 5374, mais est applicable à un nombre quelconque.

Comme le représente la figure 6, par souci de commodité, les colonnes de calculs sont numérotées de 1 à 8 et les

pointeurs I, J sont utilisés d'une grande manière comme ils seraient utilisés pour mettre en oeuvre l'algorithme de l'ordinateur. Initialement, le calcul commence avec les deux pointeurs I, J dans la colonne 1, puis I est déplacé vers la gauche colonne par colonne jusqu'à la dernière colonne du nombre x (ici la colonne 4) et, enfin, J est déplacé vers la gauche colonne par colonne jusqu'à la dernière colonne. Après chaque déplacement du pointeur I ou J, une sommation de produits croisés est obtenue pour les colonnes embrassées par I et J. (1) Quand I et J englobent un nombre pair de colonnes, n, la somme des produits croisés des colonnes englobées par I et J est obtenue. (2) Quand I et J englobent un nombre impair de colonnes, le carré de la colonne médiane est obtenu et ajouté à la somme des produits croisés des colonnes externes s'il y en a (si le nombre englobé n = 1 il n'y a pas de colonnes externes).

Cette procédure est facilement comprise en relation avec la figure 6 où I, J sont tous deux initialement à la colonne 1 et le sous-total de colonne associé est simplement  $4^2$  ou 16. Quand I est déplacé vers la seconde colonne (I = 2 et J = 1), les deux pointeurs englobent un nombre pair de colonnes et le sous-total de la colonne est  $(4 \times 7 = 28) + (7 \times 4 = 28)$  ou 56. On notera que dans chaque cas quand les produits croisés sont obtenus, deux valeurs égales telles que 28, 28 sont obtenues et les calculs peuvent être réduits en multipliant simplement le produit croisé tel que 28 par 2.

En continuant avec notre sous-programme de calcul, ensuite I est déplacé vers la colonne 3 (I = 3 et J = 1) fournissant le sous-total de colonnes associées  $(4 \times 3 = 12) + (7 \times 7 = 49) + (3 \times 4 = 12)$ . Le processus continue jusqu'à ce que I soit déplacé vers la colonne la plus à gauche puisque J soit déplacé vers la dernière colonne (I = 4, J = 4), fournissant un produit croisé associé égal à  $5 \times 5 = 25$ .

Le résultat au carré est obtenu en ajoutant simplement les colonnes.

On notera qu'à tout instant donné le processus nécessite une quantité maximale de mémoire de travail égale à deux

fois le nombre d'octets occupé par le nombre non élevé au carré  $x$  plus seulement 6 octets supplémentaires. Ainsi, l'algorithme permet un calcul d'un nombre  $x^2$  très grand en utilisant la même mémoire RAM de travail que cela est requis pour mesurer le grand nombre  $x$  plus 6 octets, mais réduit également le nombre de multiplications pour obtenir  $x^2$  de 111 bits d'environ moitié, d'environ 2100 à 1100. Ceci réduit le temps de calcul d'ensemble d'environ 25 %, soit d'environ 0,5 secondes à environ 0,365 secondes.

#### 10 C. PROTOCOLE FLEXIBLE ET FONCTIONNEMENT

Le protocole flexible est une conséquence de l'utilisation de la cryptographie à clef publique du type à signature numérique pour coder la zone de messages d'une carte magnétique. Comme cela a été décrit ci-dessus, l'approche du type signature numérique assure une excellente sécurité. En outre, le codage de la zone de messages de données en utilisant l'approche de signature numérique sépare la fonction de validation de sécurité de la fonction de message. Ceci libère le protocole quant aux limitations de programmation associées à la réalisation simultanée des fonctions de messages et de sécurité. Un exemple d'une telle contrainte se trouve dans les problèmes de séquençement exposés ci-dessus dans lesquels des cartes de clients valides sont incapables d'actionner une serrure à la suite du défaut d'utilisation d'une ou plusieurs cartes précédentes.

#### 25 1. Organisation de cartes

Comme le représente la figure 7, pour mettre en oeuvre le protocole flexible, les cartes magnétiques 32 sont utilisées et comprennent une piste magnétique 31 sur laquelle 50 octets de données sont écrits en notation hexadécimale. En se référant à la figure 8, les 50 octets de données sont divisés en un en-tête de 2 octets 101, une zone de données 102 à laquelle sont alloués 46 octets et une zone terminale 103 de 2 octets. La carte est lue de la droite vers la gauche, à partir des zéros précédant l'en-tête jusqu'aux zéros finaux. Le premier octet ou le premier octet dont on tient compte de données de la carte est un ou plusieurs octets de caractères de synchronisation dans l'en-tête qui donnent

instruction à la serrure de lire et d'analyser des données suivantes. Le second octet de données, dans l'en-tête, sert à spécifier la longueur, couramment le nombre 48, qui spécifie le nombre d'octets de zones de données et de zones terminales sur la  
 5 carte et permet un agrandissement ultérieur de la carte. Par exemple, actuellement, la longueur est choisie égale à 48 (en valeur hexadécimale S30), la longueur maximale que le microprocesseur 51 peut traiter.

La zone terminale 103 comprend des octets uniques  
 10 concernant le type de carte et une vérification de redondance longitudinale (LRC) externe. Le type de carte, le 49ème octet, spécifie actuellement l'un de six types de cartes différents : départ de fabrication ; départ de constitution ; départ de fonctionnement complet ; carte signée (établissement, programmation  
 15 ou clef) ; auto-test ; ou piste d'audit amortie. Le 50ème octet, le LRC externe d'un octet est utilisé pour vérifier que les données sont correctement lues au niveau de la serrure.

Alors que certaines cartes n'ont pas besoin d'être munies de signature, la flexibilité du protocole de l'invention  
 20 est peut être mieux représentée par les cartes - y compris les cartes-clefs et les cartes de programmation - dans lesquelles la zone de données 102 est cryptée en tant que signature numérique. Spécifiquement, et en relation avec la figure 9, le protocole de carte-clef et de programmation localise certaines informations  
 25 dans la zone de données 102 de chaque carte aux mêmes octets. Actuellement, la carte prévoit un octet pour des zones de drapeau communes, 4 octets pour un numéro d'identification de carte (ID), 2 octets pour des numéros de fréquences de zone commune, 1 octet pour un pont négatif de zone commune (ci-après), 36 octets pour le champ de message, 1 octet pour la validation LRC et 1 octet pour  
 30 divers drapeaux.

Les octets de drapeaux de zone commune spécifient une zone d'accès commune limitée. Actuellement les bits 0 à 3 permettent un accès de carte à aucune, quelques-unes, ou toutes  
 35 parmi quatre zones communes possibles à accès limité.

Le numéro d'identification (I.D) de carte comprend un

nombre à 4 octets spécifique de la clef, un parmi 4 milliards de nombres qui sont fournis dans l'ordre numérique par la console au client ou à l'employé auquel la carte est fournie.

Il faut noter que les zones communes sont les champs de formation qui sont conçus pour assurer un large accès par un grand nombre de clefs à une ou plusieurs serrures données appliquées par exemple, à des garages, des piscines, des salles d'attente publiques etc... Le numéro de séquence de zone commune est modifié automatiquement au niveau de la console de façon périodique, par exemple quotidiennement. Par exemple, avec des numéros séquentiels de chambre d'hôtel, et d'employé, si le numéro de séquence commun sur la carte est égal au numéro de serrure,  $S_C = S_L$ , la porte est ouverte, et comme cela est le cas avec des numéros de séquence d'employés de chambre d'hôtel, si le numéro de séquence commun sur la carte est supérieur au numéro sur la serrure d'une différence non supérieure au pont de séquence  $b$  ( $b \geq S_C - S_L > 0$ ) alors, non seulement la porte est ouverte, mais encore le nombre de séquences sur la carte est mémorisé dans la serrure comme ce nombre. Contrairement aux approches classiques exposées précédemment, cette technique de séquençement permet à une carte valide d'actionner un verrou indépendamment de l'utilisation ou de la non-utilisation des cartes précédentes, pour autant que la longueur de pont arbitrairement choisie ne soit pas dépassée. Comme cela a été mentionné, cette flexibilité est rendue possible du fait que l'on sépare l'actionnement du protocole de carte et de serrure de la fonction de sécurité. Le numéro de pont arbitraire  $b$  peut être 1 ou 10 ou 255 ou tout nombre qui confère la flexibilité désirée au système.

Contrairement aux numéros de séquence de chambre d'hôtel et d'employés, si le numéro de séquence commun sur la carte est inférieur au numéro dans la serrure d'une différence non supérieure au pont négatif de zone commune spécifié sur la carte  $b_C$  ( $b_C \geq S_L - S_C > \phi$ ) alors la porte est ouverte. L'accès des zones communes expire automatiquement quand la différence entre  $S_L$  et  $S_C$  dépasse le nombre de pont négatif commun  $b_C$ . Le nombre de pont négatif de zone commune est établi de façon similaire au

nombre de pont sauf que le pont négatif est spécifié dans le pont négatif de zone commune à un octet.

On considère par exemple un client et un nombre de pont négatif de zone commune égal à 10. Quand le client utilise la piscine le premier jour de son séjour la porte s'ouvre. S'il est le premier des clients de la journée à utiliser la piscine, alors le numéro de séquence sur sa carte sera supérieur au numéro de la serrure, de sorte que la serrure se mettra à jour sur le nouveau numéro de la carte. Le jour suivant, après que la serrure ait été utilisée par des clients rentrant ce jour-là, le numéro de séquence sera à nouveau avancé. Mais, notre carte de client le laissera encore entrer dans la piscine puisque, alors que sa carte a un numéro de séquence qui est inférieur au numéro de verrou, la différence est de 1, ce qui est moins que le pont négatif de -10 sur sa carte. Notre carte de client déverrouillera la piscine pendant 10 jours, pour autant que son numéro de séquence de carte est inférieur au numéro de séquence de la serrure de piscine d'une différence non supérieure au pont négatif égal à 10 sur sa carte.

Le 45ème octet dans la zone de données 102 est une vérification de redondance longitudinale (LRC) interne d'un octet qui prouve la validité de la carte. Ainsi, ce LRC interne est utilisé pour déterminer si la carte telle que déchiffrée est valide. Les 44 octets précédents sont soumis à une fonction OU Exclusif avec le LRC et le résultat zéro est requis pour que les données soient valides. Sinon, on suppose que la carte est invalide et elle est rejetée par la serrure.

Le 46ème et dernier octet dans la zone de données est utilisé pour des chose telles que la commande audio et l'indication de chute de batterie et pour spécifier si la carte est une carte d'établissement ou une carte de clef/programmation. En outre, les 2 bits d'ordre inférieur du 46ème octet sont utilisés pour un contrôle de reste quadratique. Le bit inférieur est toujours à 0 et le bit suivant est toujours à 1 de sorte que la zone de données est un nombre pair à 46 octets congruent à 2 modulo 4, ce qui facilite le déchiffrement de la carte.

#### D. PROGRAMMATION DES CARTES-CLES

##### 1. Données du champ de message.

Le champ de message à 36 octets 104 de la figure 9 communique à la serrure la/ou les fonctions qu'elle doit réaliser.

Comme cela est représenté schématiquement en figure 10, le microprocesseur et la mémoire de la serrure sont conçus pour recevoir des messages de cartes constitués de sous-messages : une ou plusieurs actions précédées d'une spécification optionnelle ou requise de zone/séquence, de nombre de serrures, et/ou de temps. Une fin EOM à 1 octet du code de message est utilisée sur la carte quand le champ de 36 octets n'est pas rempli.

Une paire zone/séquence est une zone avec un numéro de séquence associé et doit valider la plupart des actions. Le champ de message embrassera 32640 zones possibles telles que des chambres ou des suites de clients à une ou plusieurs portes, etc..

Tel qu'il est utilisé ici, le terme "zone" désigne un ensemble d'une ou plusieurs serrures liées dont toutes peuvent être ouvertes avec la même paire zone/séquence. Comme cela est représenté schématiquement en figure 12, des zones sont utilisées pour désigner un ensemble de serrures associées. A leur tour, des niveaux maîtres se rapportent à un ensemble de zones associées. Les figures 12A, 12B, 12C et 12D sont prises à partir de la figure 12 et représentent les zones et les serrures associées à trois niveaux maîtres indiqués à titre d'exemple : client (figure 12A) ; femmes de ménage (figures 12B et 12C) ; et urgence (figure 12D). Les figures sont uniquement illustratives car le domaine d'application de ce concept est beaucoup plus étendu que celui qui est représenté. Par exemple, actuellement, les serrures peuvent être programmées pour répondre à jusqu'à neuf zones ou niveaux maîtres. L'utilisation de niveaux maîtres dans les serrures classiques est limitée à plusieurs serrures fixes désignées ou groupement de serrures et chaque serrure est limitée à une sélection choisie dans ce groupe. Toutefois, en utilisant le présent protocole, une très large sélection de niveaux (environ 32640) est disponible.

Spécifiquement, en ce qui concerne le protocole de zone, un octet inférieur de zone égal à zéro n'est pas autorisé

sur une carte ; les 128 telles zones possibles sont réservées à une utilisation de serrure. Les 15 bits inférieurs du champ de zones de 16 bits spécifient la zone elle-même. Il y a ainsi 32640 zones possibles spécifiées par les 15 bits. Chaque zone en utilisation est associée à un numéro de séquence en cours. L'organisation des types et des numéros de portes est définie par le service de gestion sur chaque site. Alors qu'une chambre d'hôtel avec une porte représente une zone d'une serrure, la zone d'urgence est constituée de la plupart ou de toutes les serrures de l'hôtel ou du système. Dans les deux cas, un numéro de séquence unique est associé avec chacun.

Le bit 14, le bit d'ordre le plus élevé dans la zone (le second bit d'ordre le plus élevé dans le champ de zone), spécifie si la zone est destinée à un accès de client ou d'employé. Si ce bit est établi, la zone est considérée comme étant une zone d'employés. Si le bit est zéro, la zone est considérée comme une zone de clients.

Comme cela a été mentionné ailleurs, la première zone de toutes les serrures est la zone d'urgence. Elle n'est jamais enlevée et ne comprend pas de compteur de fois unique. Une clef d'urgence valide peut ouvrir toutes les serrures pourvu qu'il existe une zone d'urgence unique, ou s'il y en a plus, des paires de niveaux d'urgence Zone/Séquence, tous les ensembles sont sur la clef d'urgence. Si le bit supérieur (le bit 15) de la zone d'urgence est établi, ceci indique une ouverture prioritaire de verrou, toutes les serrures sont programmées pour s'ouvrir à un instant quelconque indépendamment de la position de leur verrou sur la porte ou indépendamment de la présence d'un état de haute sécurité. Toutefois, si le bit de priorité de verrou n'est pas établi, alors la carte ne peut ouvrir la porte si elle est verrouillée par un verrou ou tout autre état de sécurité.

Les zones de clients sont également soumises à un traitement spécial. Seule une mise à jour de séquence de zone de clients remettra à zéro un état de haute sécurité (exposé par ailleurs) et alors qu'il peut y avoir des zones de clients multiples programmées dans une serrure, une seule peut être active

à un instant particulier - les autres sont bloquées. La mise à jour de la séquence d'une zone de clients en fait la zone de clients active et bloque toutes les autres. Une zone de clients bloquée peut également être rendue active par l'utilisation d'une opération de remise à zéro du blocage.

Le bit 15, le bit d'ordre le plus élevé de chaque champ de zone sur une carte spécifie un déblocage prioritaire d'un verrou. Quand le bit 15 est à 1, la clef ouvrira la porte même si un état de haute sécurité existe ou même si le verrou a été tiré de l'intérieur, comme cela a été représenté pour la clef d'urgence ci-dessus. Quand un bit 15 sur une zone est à 0, la carte n'ouvrira pas la porte si un état de haute sécurité existe (à moins que l'action soit "Set High Security/Open" (mettre haute sécurité/ouvrir) exposé ci-dessous) ou bien si le verrou a été enlevé de l'intérieur.

Le numéro de Séquence à deux octets est appairé avec le numéro de Zone pour valider la plupart des actions que la serrure peut effectuer. Quand une paire Zone/Séquence valide une action "ouvrir la porte" le logiciel mémorisé dans une ROM de la serrure compare la paire aux Zones et Séquences présentement mémorisées dans la serrure. On se référera à l'exemple d'organisation de mémoire de serrure de la figure 11. S'il trouve qu'une Zone a été programmée dans la serrure, il compare alors les Séquences. Si le numéro de Séquence est égal au numéro de Séquence déjà dans la serrure dans la Zone spécifiée, alors la serrure exécutera l'action désirée. Si la Séquence lue à partir de la carte est supérieure à la Séquence dans la serrure dans cette zone spécifiée et que la différence entre les deux n'est pas supérieure à la valeur de pont, alors la serrure exécute également l'action désirée et, si l'action validée est l'une des cinq actions de clef (ouvrir, mettre haute sécurité/ouvrir, ouvrir une fois, déverrouiller ou reverrouiller) ou est une action de programmation de séquences de mise à jour et que le reste du message et du champ de message est valide, la fonction désirée réalisée et le numéro de Séquence est mis à jour. Ceci signifie que le numéro de séquence de la carte remplace le numéro de séquence précédemment programmé

dans la serrure. De cette façon, de vieilles clefs sont automatiquement invalidées chaque fois qu'une nouvelle clef est utilisée sur chaque serrure dans chaque zone.

On notera toutefois que seules les actions spécifiées mettront à jour la séquence de serrure. Si la première action n'est pas l'une des actions spécifiées, la Séquence ne sera pas mise à jour par ce message. En outre, plusieurs paires Zone/Séquence peuvent être aussi spécifiées sur une carte unique. Egalement, il faut noter que la présente capacité de la serrure permet de prévoir jusqu'à huit paires Zone/Séquence sur chaque serrure. Si moins de huit sont spécifiées, certaines peuvent être conditionnées par une option spéciale Temps. Si deux ou plus paires Zone/Séquence sont spécifiées et que l'une s'adapte à la serrure correspondante alors qu'une autre mettrait à jour la séquence, alors la mise à jour prend place indépendamment de l'accord au niveau de l'autre zone. S'il existe au moins deux paires Zone/Séquence sur une carte qui mettraient à jour les séquences correspondantes dans une serrure, elles sont toutes mises à jour.

Le numéro de serrure (Lockno) est un nombre à deux octets qui est assigné par la console à chaque serrure et qui ne concerne en aucune façon le numéro de chambre sur lequel la serrure est installée et identifie de façon spécifique la serrure.

La spécification temporelle (Timespec) agit quand une carte optionnelle horloge/calendrier est prévue pour une serrure et permet à des cartes d'être valides seulement à des dates et heures particulières ou pendant certains jours ou les deux.

La carte horloge/calendrier est une carte optionnelle pour chaque serrure. Quand elle est connectée, elle assure une capacité de sécurité accrue : des cartes peuvent avoir une validité limitée seulement pendant certaines dates et heures spécifiques ou certains jours ou les deux et des transactions sont fixées dans la carte. Deux codes optionnels (Opcodes) peuvent être prévus pour régler la date, le jour et l'heure convenables dans la puce horloge/calendrier. D'autres Opcodes sont prévus pour valider et limiter des actions de carte.

Des spécifications temporelles peuvent être écrites sous forme de message sur les cartes pour limiter la validité d'une opération à certaines dates ou heures. La serrure comparera le jour/date/heure dans son propre horloge/calendrier aux instants sur la carte pour déterminer la validité de l'opération.

Les spécifications temporelles peuvent consister en un ou plusieurs Opcodes de spécifications temporelles, suivis chacun d'un ou plusieurs opérandes jour/temps. Normalement, un seul Opcode de spécification temporelle sera utilisé. On peut en appeler un second si la partie d'opérande de la spécification temporelle est plus longue que la longueur de 15 octets que cet Opcode peut spécifier. Dans ce cas, un second Opcode est utilisé pour continuer la spécification temporelle.

#### E. ACTIONS DES CARTES

Une carte peut réaliser deux actions : programmer la serrure au moyen d'une ou plusieurs fonctions et ouvrir la serrure. Les différents types possibles d'actions de clef comprennent : ouverture simple (toute serrure avec une combinaison s'adaptant au niveau maître spécifié), placer haute sécurité/ouvrir ; déverrouiller (créer une voie de passage) ; reverrouiller (une voie de passage) ; et ouvrir une fois (pour une personne d'entretien ou de service, etc). Les actions de programmation comprennent : régler l'horloge à la date/heure/jour ; effacer les zones communes ; bloquer un ou plusieurs niveaux maîtres de clef ; remettre à zéro le blocage ; mettre à jour le numéro de séquence de serrure à la valeur en cours ; ajouter Zone (accepter des clefs supplémentaires) ; et supprimer Zone. Ces actions sont présentées ci-après.

#### 1. Actions d'ouvertures

##### a. Ouverture

Ce sous-message de données ouvre la serrure si les nombres de validation optionnels Lockno et Timespec s'adaptent aux données de la serrure et si la zone/Séquence de validation fait un pont ou s'adapte.

Des exceptions comprennent les cas suivants : (1) si le pêne dormant de la serrure est classé, le bit de priorité de

pêne dormant dans la Zone doit être classé ou bien la porte ne pourra être ouverte par la carte ; (2) Si Haute Sécurité est placée et que la validation se fait par une zone de clients qui ne met pas à jour le numéro de séquence, le bit de priorité de verrou dans la zone doit être placé ou bien la porte ne pourra être ouverte par la carte ; et (3) si la Zone de validation est bloquée et ne met pas à jour le numéro de Séquence, la porte ne pourra être ouverte par la carte.

Une action d'ouverture met à jour les séquences associées à toutes les zones de validation qui forment un pont. Une mise à jour valide de séquence remet à zéro tout blocage au niveau de la zone de mise à jour ainsi que, si la zone mise à jour est une zone de clients (bits 14 à 0), remet à zéro le verrou logique (voir Haute Sécurité ci-dessous).

b. Etablir Action d'Ouverture Haute Sécurité

Cette action est identique à l'action d'Ouverture, sauf que la première action de la carte est de placer un verrou "logique". Une fois le verrou placé, les seules cartes qui ouvriront la serrure sont celles munies d'un bit de priorité de verrou ou d'une action d'établissement de Haute Sécurité/Ouverture ou celles qui mettent à jour la séquence associée à une zone de clients (bits 14 à 0). Alors que toute carte peut établir l'état de Haute Sécurité, seule une clef de client (bits de zone 14 à 0) peut le remettre à zéro lors d'une mise à jour de séquence.

c. Action de Déverrouillage

Cette clef amène une porte à agir comme une voie de passage ouverte jusqu'à ce qu'une clef de reverrouillage soit utilisée pour un reverrouillage. Les exceptions comprennent : (1) si le verrou de la serrure est placé, le bit de priorité de verrou dans la Zone doit être placé ou la porte ne pourra être ouverte par la carte ; (2) si Haute Sécurité est établi et qu'une validation se fait par une zone de clients qui ne met pas à jour le numéro de séquence, le bit de priorité de verrou dans l'octet de niveau maître doit être établi ou bien la porte ne pourra être ouverte par la carte ; (3) si la zone de validation est bloquée et ne met pas à jour le numéro de séquence, la porte ne pourra être

ouverte par la carte.

d. Action de Reverrouillage

Cette clef reverrouille une porte agissant comme une voie de passage et met à jour les séquences associées à toutes les zones de validation ayant besoin d'une mise à jour, pourvu que les autres préconditions à la mise à jour d'une séquence énumérées dans l'action d'ouverture soient satisfaites.

e. Action d'Ouverture pour Une Foix

Cette clef ouvre une serrure une seule fois. Les conditions d'ouvertures sont les mêmes que pour l'action d'Ouverture sauf que : (1) le compteur qui est dans l'opérande Une Foix doit être plus élevé que le compteur d'un octet dans la serrure correspondant à la zone qui ouvrirait la serrure ; et (2) s'il existe une horloge dans la serrure, une durée de validation requise doit être valide. Toute remise en séquence nécessaire est exécutée avant de valider le compteur Une Foix (sur une clef qui remet en séquence, le compteur est automatiquement valide puisse que mettre à jour la séquence remet à zéro le compteur Une Foix de serrure pour cette zone).

Si la serrure se valide (indépendamment du fait qu'elle soit ouverte ou non) alors le compteur dans la serrure est rendu égal au compteur sur la clef, empêchant ainsi une réutilisation de la clef, et empêchant également une utilisation de toute clef Une Foix fournie avant celle-ci (avec des comptages plus faibles dans leurs opérandes). Le compteur dans la serrure est séquencé même si la porte n'est pas ouverte (en raison du verrou placé ou d'une absence de priorité, par exemple, ou d'un blocage de la zone de validation).

Il existe un octet de compteur par zone dans la serrure, sauf au niveau de la zone d'urgence (la première zone ajoutée par la carte d'établissement, de sorte que la zone ne peut être utilisée pour valider cette clef).

2. Actions de Programmation de Cartes

a. Opération d'Etablissement d'Horloge

L'opération d'Etablissement d'Horloge est validée en préfaçant l'opération sur la carte par toute Zone/Séquence qui se

trouve également dans la serrure. L'horloge de la serrure est placée à la date, heure et jour de la semaine qui sont spécifiés dans l'opérande.

b. Opération d'obtention d'heure de Terminal Portable

5 Si une serrure peut communiquer avec un terminal portable dans des buts de suivi de vérification, alors un terminal portable peut également être utilisé pour établir la date, l'heure et le jour dans la serrure.

10 Ceci fonctionne de la façon suivante : le terminal portable décharge la date, l'heure et le jour de la semaine ainsi qu'un programme de communication de serrure à partir de la console ; le terminal portable est connecté à la serrure ; la carte d'obtention d'heure est amenée à passer dans le lecteur de  
15 cartes de la serrure ; la serrure valide la carte par rapport à la Zone/Séquence sur la carte, ainsi que par le compteur une fois sur la carte à cette zone ; la serrure répond en lisant la date, le jour et l'heure de la semaine à partir du terminal portable par l'intermédiaire de son accès série.

c. Opération d'Etablissement de Zone Commune

20 Cette opération convertit une serrure en accès de Zone Commune et lui donne une Séquence de Zone Commune pour répondre et optionnellement des heures pour une accessibilité en Zone Commune. Cette opération nécessite que le message contienne le Lockno valide et toute Zone/Séquence valide dans la serrure. Une  
25 spécification temporelle est alors requise (bien qu'elle soit seulement utilisée par des serrures avec horloge).

Les niveaux d'accès à une Zone Commune de serrures sont établis pour s'adapter aux quatre drapeaux de zone commune dans le champ de drapeaux de la carte. Si aucun des quatre  
30 drapeaux n'est établi, le drapeau d'accès de Zone Commune non limitée de serrures est établi pour indiquer que toute clef de site valide avec un numéro de séquence de Zone Commune valide ouvrira la serrure.

Le numéro de séquence de Zone Commune est remplacé par  
35 le numéro de séquence de zone commune sur la carte. L'établissement de Zone Commune comprend également l'option

consistant à régler un ensemble d'heures pendant lequel l'accès commun sera autorisé et/ou un ensemble de jours pendant lequel l'accès commun sera autorisé (si les deux sont spécifiés, alors les deux doivent être vrais pour que la serrure autorise un accès commun).

5        d. Opération d'Effacement de Zone Commune

L'opération d'Effacement de Zone Commune supprime tous les accès communs à une serrure. Cette opération requiert que le message contienne une quelconque Zone/Séquence valide dans l'horloge. Tous les drapeaux d'accès de Zone Commune de serrures et les informations de Séquence et de Temps sont effacés par cette opération.

10       e. Opération de Blocage

L'opération de Blocage bloque les zones spécifiées dans l'opérande. Elle est validée par la Zone/Séquence spécifiée.

Un blocage peut être annulé de deux façons :

Une clef qui met à jour la Séquence associée à une Zone dans une horloge remettra à zéro le Blocage au niveau de la Zone mise à jour. (S'il s'agit d'une Zone de clients, la procédure de mise à jour met également automatiquement un blocage sur toutes les autres Zones de clients).

Une carte de Remise à Zéro de Blocage (voir opération de Remise à Zéro de Blocage) remettra à zéro les zones spécifiées qui ont été bloquées.

20       f. Opération de Remise à Zéro de Blocage

Cette carte remet à zéro le blocage installé au moyen d'une carte d'opération de Blocage remettant à zéro les blocages au niveau des zones spécifiées dans l'opérande, validant la carte à l'encontre de toute paire Zone/Séquence dans la serrure.

25       g. Opération de Mise à Jour de Numéro de Séquence en Valeur  
Présente

La mise à jour de Séquence est la seule carte de programmation à exécuter des sous-programmes de mise à jour-séquence dans la serrure. Elle diffère d'une clef d'Ouverture (action d'Ouverture) principalement en ce qu'elle ne déverrouille ou n'ouvre jamais une porte. Son objet est seulement de mettre à jour

associé. Ceci assure une compatibilité vers le haut et vers le bas entre des serrures et des cartes anciennes et nouvelles.

Par exemple, si de nouvelles serrures sont ajoutées ou que des serrures sont modifiées pour avoir des capacités qui n'existent pas dans des serrures existantes, les serrures  
 5 anciennes pourront néanmoins être actionnées par les cartes-clefs contenant les nouveaux sous-messages en dépit de l'incapacité des anciennes serrures à comprendre et à mettre en oeuvre les nouveaux sous-messages. Cette compatibilité vers le bas, entre les  
 10 nouvelles cartes et les anciennes serrures et entre les anciennes et les nouvelles serrures, existe du fait que, quand la serrure ancienne ne présente pas la capacité de comprendre ou de mettre en oeuvre le nouveau sous-message, elle peut tout simplement sauter la longueur prédéterminée du nouveau sous-message vers le message  
 15 suivant qui se trouve dans ses capacités de programmation.

Le système est aussi compatible vers le haut en ce que les nouvelles serrures mettent en oeuvre facilement toutes les instructions des anciennes serrures contenues dans des anciennes cartes. Dans la mesure où de nouvelles serrures n'ont pas besoin  
 20 d'être programmées pour mettre en oeuvre un sous-message ancien particulier, comme les anciennes serrures, elles sautent tout simplement le sous-message particulier jusqu'au sous-message suivant qu'elles sont programmées pour mettre en oeuvre.

En bref, tant que les anciennes et nouvelles cartes  
 25 comprennent les Opcodes les unes des autres, une compatibilité complète vers le bas aussi bien que vers le haut existe, permettant un usage mixte d'anciennes et de nouvelles serrures, de nouvelles cartes avec d'anciennes serrures, et inversement.

## 2. Clef Une Fois

Une autre retombée directe de l'utilisation d'un  
 30 protocole flexible réside dans la capacité de fournir des clefs dites Une Fois qui permettent l'entrée dans une zone désignée 2 à 9 (excluant l'urgence bien entendu) de personnels de livraison, tel qu'un fleuriste et analogue. Comme cela est représenté en  
 35 figure 11, la table de vérification dans chaque serrure comprend un champ Une Fois qui est validé par Zone et Séquence et

la séquence dans une serrure de sorte que les séquences précédentes sont bloquées sans avoir également à ouvrir la serrure simultanément.

Si la clef d'Urgence devait être modifiée par suite de la perte ou du vol de l'une d'elles, une carte de mise à jour de Séquence serait placée dans chaque serrure de l'hôtel par un employé de bas niveau dont on a besoin seulement qu'il l'utilise sur toutes les serrures, qu'il ne la vole pas lui-même, ou qu'il n'en fasse pas de copie (puisque'il n'ouvre pas la porte, il n'y a pas de risque de vol ou de perte). Les clients ne seront pas perturbés par le bruit de l'ouverture de leur porte simplement dans un but de mise à jour de sa séquence.

#### h. Opération d'AJout de Zone

L'opération d'AJout de Zone ajoute les paires d'opérandes de Zone/Séquence à la serrure. Si une serrure comprend déjà une Zone à ajouter, ou si toute la mémoire de zone de la serrure est déjà en utilisation, le champ de message complet est ignoré et les voyants sont amenés à clignoter pour signaler un état d'erreur. Une paire quelconque Zone/Séquence est requise pour validation.

#### i. Opération de Suppression de Zone

Cette opération supprime de la serrure les Zones spécifiées dans l'opérande. Toutefois, la Zone d'urgence ne peut être supprimée d'une serrure ; le fait d'essayer de le faire invalide toute la carte.

### P. AUTRES CARACTERISTIQUES DE PROTOCOLE FLEXIBLE

#### 1. Compatibilité Haut/Bas

Le présent protocole flexible est conçu pour que des sous-messages individuels dans le champ de messages à 36 octets comprenant les sous-messages Zone, Séquences, Lockno (numéro de serrure) Timespec (spécifications temporelles) et Actions, comprennent chacun un Opcode (code d'opération) qui occupe une longueur spécifiée selon son type et le type d'opérande. La longueur ainsi que le type d'opérande est spécifiée par l'Opcode. Ainsi, en spécifiant sa propre longueur et la longueur de l'opérande, l'Opcode spécifie complètement la longueur totale du sous-message

optionnellement par Timespec. Chaque carte Une Fois contient une zone et une séquence particulières et contient également des numéros Une Fois fournis séquentiellement. Chaque serrure est programmée pour s'ouvrir si le numéro de la séquence sur la carte Une Fois est supérieur au numéro de la séquence Une Fois de la serrure et remplace ensuite son numéro de séquence Une Fois par le numéro de la carte. Ainsi, chaque nouvelle utilisation d'une carte Une Fois convenablement mise en séquence bloque toutes les cartes Une Fois précédentes, qu'elle soient validement délivrées ou non.

Par exemple, si la réception de l'hôtel fournit une première carte Une Fois pour la chambre 201 à un fleuriste, fournit ensuite une seconde carte à un télégraphiste puis fournit une troisième carte à un livreur d'épicerie et que le livreur d'épicerie va directement à la chambre 201 alors que le fleuriste et le télégraphiste retardent leur livraison, l'utilisation de la troisième carte bloque non seulement cette carte mais également toutes les cartes précédentes même si ces cartes précédentes n'ont pas été utilisées.

Une serrure contenant la carte d'option supplémentaire horloge/calendrier peut en outre limiter la carte à une couverture de Timespec, par exemple des périodes temporelles particulières. En outre, des cartes Une Fois peuvent être établies pour l'un quelconque ou tous les niveaux de 2 à 9 d'une serrure individuelle, seulement sous la condition qu'elles soient convenablement délivrées en accord avec leurs séquences présentes pour les différents niveaux.

### 3 Accès multiples ; Programmation et Action de combinaison.

La possibilité de programmer des sous-messages multiples sur une carte donnée transforme effectivement la carte en un porte-clef sur lequel chaque sous-message représente une clef.

En outre, des fonctions de programmations et des actions de clefs peuvent être combinées sur une carte et peuvent être validées par la même zone ou des zones différentes.

### G CIRCUIT ELECTRONIQUE DE COMMANDE DE SERRURE.

Comme cela est représenté schématiquement en figure

10, le circuit de commande principal 50 de la serrure électronique 30 comprend un microprocesseur 51 et cinq parties principales qui présentent des interfaces avec l'ordinateur : un circuit d'alimentation 52 ; un circuit de réveil 53 ; des entrées de serrure 54 ;  
5 des sorties de serrure 56 ; et une interface 57 pour une carte d'option supplémentaire.

La serrure est conçue pour fonctionner avec des micro-ordinateurs tels que le HD6305V0 ou le 68HC05C4, qui sont sensiblement identiques, comprennent 4096 octets de ROM et 192  
10 octets de RAM et quatre accès d'entrée/sortie (I/O) parallèles : PA 0-7, PB 0-7, PC 0-7 et PD 0-7. Le circuit d'alimentation 52 représenté au coin inférieur gauche de la figure comprend une source d'alimentation de six volts 58, de préférence sous la forme de piles au lithium ou alcalines qui sont connectées par l'intermédiaire d'une fiche 59 au micro-ordinateur 51 et aux autres parties  
15 du circuit de commande. Quand il est en sommeil (l'horloge ne fonctionnant pas), le micro-ordinateur 51 fonctionne à très faible puissance, de l'ordre de 10  $\mu$ A (micro-ampère). Le circuit d'alimentation 52 est divisé en 5 bus d'alimentation VBATT, VW<sup>+</sup>, VM<sup>+</sup>, VB<sup>+</sup>, et VS<sup>+</sup> dans le but d'assurer une longue durée de vie à  
20 la source d'alimentation 58 pour maintenir le contenu de la mémoire RAM du micro-ordinateur quand les batteries sont enlevées ou usées. Ceci est effectué essentiellement pour maintenir l'enregistrement du suivi de vérification du micro-ordinateur. On  
25 notera que comme un "ordinateur" contient un "processeur" les deux termes peuvent être utilisés ici parfois l'un pour l'autre. En particulier, le micro-ordinateur 51 peut être appelé microprocesseur 51 quand c'est la fonction de processeur qui est discutée ou mise en lumière.

Le bus d'alimentation VBATT alimente directement un transistor 61 qui est connecté à un condensateur 62 de grande capacité pour charger le condensateur à la tension de batterie. Actuellement, un condensateur de 15 000  $\mu$ F (microfarad) 62 est  
30 utilisé. Comme cela sera décrit ci-dessous, le condensateur 62 est utilisé pour fournir des impulsions à un solénoïde 78 pour effectuer le blocage et le déblocage du pêne 33 de la figure 4.

33

Le deuxième bus,  $VM^+$ , alimente le micro-ordinateur 51, le circuit de réveil 53, et les circuits intégrés CMOS à basse puissance tels que 66, 67 et 68. Le bus  $VM^+$  est alimenté à partir d'un condensateur de forte valeur 69 pour maintenir l'alimentation vers le microprocesseur 51 et sa mémoire vive (RAM) pendant au moins dix heures dans le cas où les batteries auraient été enlevées ou fonctionneraient mal.

Le troisième bus,  $VW^+$  alimente le commutateur de réveil 71 pour activer sélectivement le micro-ordinateur 51 pendant une durée prédéterminée pour lire et mettre en oeuvre les instructions de carte et actionner la serrure 30. Pendant un état d'enlèvement ou de mauvais fonctionnement de batterie, il est nécessaire de maintenir le microprocesseur dans son état de repos, "de sommeil", pour minimiser la consommation de puissance et augmenter ainsi la durée pendant laquelle le condensateur 69 peut maintenir l'alimentation sur le microprocesseur. Le circuit de réveil 53 a une configuration propre à empêcher l'actionnement ou le réveil du microprocesseur 51 pendant cette durée.  $VW^+$  n'est pas associé à un condensateur de maintien et est isolé par une diode de l'autre bus (l'émetteur du transistor 61 agit comme diode dans ce but).

Le bus  $VS^+$  est utilisé pour commander les dispositifs à fort courant qui ne comprennent pas de commutateurs séparés (qui ne sont pas commandés individuellement tels que, par exemple, le lecteur de carte de serrure et le circuit détecteur de baisse de batterie. Le bus  $VS^+$  lui-même est connecté par une ligne ENAB  $VS^+$  à une sortie PAD du micro-ordinateur pour commuter la tension de bus en arrêt/marche.

Enfin, le bus  $VB^+$  alimente les LED d'état 36, le vibreur 40 et le relais 80.

Comme cela a été mentionné, le fonctionnement du microprocesseur 51 est initialisé par le circuit de réveil 53 du fait que l'on insère la carte 32 dans le lecteur de carte de la serrure. Tandis que la carte 32 est poussée dans la fente 38 du lecteur, figure 4, le commutateur de réveil 71 est fermé pour appliquer la tension à partir du bus  $VW^+$  à l'entrée IN-A de la

moitié supérieure 66 d'un circuit monostable 65. Le circuit monostable supérieur 66 fournit une impulsion constante d'une milliseconde quand il est actionné et excite l'entrée RESET du micro-ordinateur pour remettre à zéro le microprocesseur éveillé.

5 Le circuit inférieur 67 du monostable 65 est conçu pour avoir une seconde durée, par exemple trente secondes, ce qui est plus long que la plus grande durée pendant laquelle le microprocesseur est actif avant de revenir à son état de repos.

Les interconnexions représentées entre les circuits monostables supérieur et inférieur et le microprocesseur 51 ont

10 une configuration telle que, quand le commutateur de réveil 71 fournit une impulsion au circuit monostable supérieur 66, l'impulsion d'une milliseconde sur la broche de sortie Q est fournie à la broche RESET du microprocesseur et est également appliquée à

15 l'entrée IN-A du circuit monostable inférieur 67, déclenchant ainsi le circuit inférieur pour produire son impulsion de 30 secondes sur sa sortie Q. Cette dernière impulsion est réappliquée à la broche d'entrée ENAB du circuit monostable supérieur pour invalider le circuit supérieur, c'est-à-dire pour empêcher le

20 circuit supérieur de se déclencher à nouveau. Le circuit monostable supérieur 64 est invalidé pendant la seconde durée de 30 secondes de l'impulsion de sortie sur la moitié inférieure, c'est-à-dire tant que le circuit inférieur est encore en temporisation et le microprocesseur ne peut pas être remis à zéro par

25 inadvertance pendant cette période.

Juste avant que le microprocesseur ne revienne à son état de repos, il fournit une impulsion de sortie ENAB 30/SEC TIMER (valider le temporisateur de 30 secondes) par l'intermédiaire de sa sortie PC6 qui est appliquée à l'entrée ENAB du

30 circuit monostable inférieur 67 pour remettre à zéro ce circuit qui revalide à son tour le circuit monostable supérieur 66.

En résumé, alors, le circuit de réveil 53 assure trois actions importantes. Premièrement, le circuit monostable supérieur 66 active ou remet à zéro le microprocesseur 51 quand une carte

35 est poussée dans le lecteur de la serrure. Deuxièmement, le circuit monostable inférieur 67 invalide le circuit supérieur à

l'encontre d'opérations de remise à zéro supplémentaires pendant une durée prédéterminée après cette opération de remise à zéro initiale pour permettre un fonctionnement ininterrompu du microprocesseur. Troisièmement, le microprocesseur lui-même assure le  
5 fonctionnement prioritaire de cet état d'invalidation à la fin d'un cycle de fonctionnement. En conséquence, la fermeture du commutateur de réveil 71 (par l'insertion d'une carte) peut activer le circuit de réveil 53 pour remettre à zéro le microprocesseur 51 pour commencer un autre cycle de fonctionnement  
10 ou pour achever la survenance peut probable d'une opération parasite.

Les entrées de serrure 54 comprennent une interface de lecteur de carte 74 entre le lecteur de carte de la serrure et le microprocesseur 51. Une bascule 76 mémorise temporairement les  
15 données incidentes pour laisser plus de temps pour obtenir les bits, de sorte que ceci peut être fait en la durée d'un bit ultérieurement.

Le pêne 33 (figure 4) est actionné par un poussoir sollicité magnétiquement (non représenté). Le solénoïde 78 (figure  
20 10), reçoit des impulsions de façon réversible en déchargeant le condensateur 62 par l'intermédiaire d'un transistor de puissance 79 sous la commande du relais 80. Dans son état normal non actif, le relais 80 établit la polarité du solénoïde 78 pour déverrouiller la porte. Quand il est actionné par une impulsion  
25 DIR à partir de la sortie PA3 du micro-ordinateur, le relais 80 inverse la polarité pour libérer le solénoïde pour reverrouiller la porte.

Puisque la porte n'est pas reverrouillée automatiquement, il est très important pour le micro-ordinateur de savoir  
30 quand le levier 41 a été actionné ou libéré de sorte qu'il peut effectuer un actionnement en impulsion inverse du poussoir pour le libérer et reverrouiller la porte et empêcher ainsi une entrée non autorisée. Cette fonction de protection est réalisée par un commutateur optique 85 qui est monté dans la serrure 30 et  
35 comprend une diode photoémettrice dans l'infrarouge 81 et un phototransistor 82 qui sont connectés par une fiche 83 au micro-ordinateur. La sortie PC5 du micro-ordinateur 51 commande l'actionnement

d'un amplificateur 90 appliquant une impulsion de validation sur une ligne ENAB OPTO SW (valider commutateur optique) pour activer la LED 81. La LED 81 et le transistor 82 sont disposés de sorte que le rayonnement infrarouge en provenance de la LED dirigé vers le phototransistor est normalement interrompu par le levier 41. Toutefois, quand le levier est amené à pivoter pour ouvrir la porte, il est enlevé du trajet du rayonnement infrarouge et le rayonnement incident amène le transistor 82 à produire un signal de sortie qui est appliqué à l'entrée PD1 du micro-ordinateur, amenant le micro-ordinateur à alimenter le relais 80 pour déconnecter le poussoir du levier 41. Le commutateur de verrou 86 surveille simplement la poussée du verrou 34, figure 4, sur la serrure et envoie cette information d'état au microprocesseur en PDO.

Le circuit de sortie de serrure 56 comprend les sorties PA1 à PA3 pour réaliser l'actionnement sus-mentionné du solénoïde. En outre, les sorties PA4 à PA6 sont utilisées pour éclairer les LED d'état 36 et la sortie PC7 est utilisée pour provoquer l'actionnement du vibreur 40.

La tension de charge appliquée au condensateur 62 par le transistor 61 est surveillée par un conducteur LOW BATT SENSE (détection de chute de batterie) connecté à l'entrée inverseuse du circuit de comparateur 72 qui a une configuration très similaire à celle d'un amplificateur opérationnel. Une diode Zener 87 fournit une tension de référence stable de par exemple 3,3 volts pour l'entrée non inverseuse du comparateur 72. La tension de charge sur la ligne LOW BATT SENSE est appliquée à l'entrée non inverseuse par l'intermédiaire d'un diviseur de tension 89 pour appliquer une tension à l'entrée inverseuse qui est supérieure ou égale à la tension à l'entrée de référence quand la tension de charge est supérieure ou égale à un niveau de seuil désiré (tension de batterie minimale). Ainsi, la sortie du comparateur 72 est appliquée à l'entrée PD2 du microprocesseur et est utilisée pour détecter un état de batterie insuffisante vrai ou faux.

En fait, la sortie est utilisée de deux façons différentes. D'abord, elle est utilisée pour surveiller à tout

instant donné une charge sur le condensateur 67 de sorte que le microprocesseur 51 peut maintenir le condensateur dans un état de pleine charge. Ceci assure un fonctionnement instantané du solénoïde quand une carte est poussée dans le lecteur de la serrure. Deuxièmement, le temps nécessaire pour charger le condensateur 62 fournit une indication de l'état de charge de la batterie. Une durée de charge de 5 RC où RC est la constante de temps associée à la résistance 64 et au condensateur 62, fournit normalement une charge à 99 % sur le condensateur en utilisant une batterie normalement chargée. Ainsi, si la durée de charge déterminée par le micro-ordinateur 51 dépasse 5 RC, un état de batterie insuffisant est détecté et les piles doivent être remplacées.

#### H. CARTE D'OPTIONS SUPPLEMENTAIRES

Le schéma de la figure 13 représente une carte d'options supplémentaires optionnelle horloge/calendrier 105. Cette carte est enfichée dans la carte de commande principale 50 au moyen de l'interface de carte d'options supplémentaires 57 et ajoute des caractéristiques et des possibilités supplémentaires à la serrure 30.

L'interface de carte d'options supplémentaires 57 n'est pas spécialisée en ce que plusieurs types distincts d'options, incluant, sans limitation, une carte d'option horloge/calendrier, une interface infrarouge bidirectionnelle, et une interface élévatrice peuvent être enfichés dans la carte de circuit principal 50 sans changement quelconque à cette dernière.

La carte d'option horloge/calendrier 105 est constituée de quatre parties : un circuit d'alimentation 106 ; une horloge/calendrier/RAM CMOS 107 ; un numéro de série de site 108 et une interface série 109.

Chaque carte optionnelle est alimentée à partir du circuit de commande principal 50 par l'intermédiaire des conducteurs d'alimentation de carte optionnelle VBATT et VS<sup>+</sup>. Sur la carte d'options supplémentaires horloge/calendrier, VBATT est partagé en deux bus VB<sup>+</sup> et VC<sup>+</sup>, qui sont isolés par des diodes 110 et 111. VB<sup>+</sup> est alimenté seulement si VBATT est alimenté,

c'est-à-dire quand les piles 58 sont enfichées dans la carte du circuit principal.  $VC^+$  comprend une capacité de maintien importante (1 farad) 112 pour maintenir une alimentation de sécurité vers la RAM CMOS d'horloge/calendrier 107 même si les piles sont enlevées pendant dix heures ou plus. Le bus d'alimentation  $VS^+$  est validé par le micro-ordinateur 51 par l'intermédiaire du transistor 70 sur la carte de circuit principal et est coupé quand le micro-ordinateur est en sommeil.

Le circuit RAM CMOS horloge/calendrier 107 utilise un circuit intégré commercialement disponible 113 pour fournir des fonctions temporisées pour la serrure et pour marquer et mémoriser la date et l'heure dans jusqu'à neuf entrées de suivi de vérification dans ses 50 octets de RAM CMOS.

La puce RAM d'horloge/calendrier est normalement dans un mode de repos quand la serrure est en sommeil, en raison du fait que le signal  $VS^+$  à bas niveau amène la broche STBY à être à bas niveau. Quand le micro-ordinateur "s'éveille" il amène  $VS^+$  à haut niveau, validant les autres broches d'entrée/sortie (I/O) de la puce horloge/calendrier, du circuit de numéro de série de site 108 et de l'interface série 109. Le conducteur PA 7 de l'interface de carte d'options supplémentaires 57 choisit ou bien la puce RAM horloge/calendrier quand PA 7 est à haut niveau ou bien le circuit de numéro de série de site quand PA 7 est à bas niveau. Les conducteurs PC 0 à PC 3 fournissent des lignes de commande supplémentaires pour la puce RAM horloge/calendrier et des conducteurs PB 0 à PB 7 fournissent des adresses et des données à la puce RAM d'horloge/calendrier et des données pour le circuit de numéro série de site.

Des portes 114 et 115 inhibent une interruption externe (OBIRQ) vers le micro-ordinateur quand les piles sont enlevées, en raison du fait que  $VB^+$  passe à bas niveau invalidant la porte ET 115. Cette caractéristique est analogue à l'invalidation du commutateur de réveil 71 sur la carte principale quand les piles sont enlevées par suite du fait que le bus d'alimentation  $VW^+$  passe à bas niveau. Dans les deux cas, le but est de ne pas autoriser le micro-ordinateur à s'éveiller quand les batteries

sont enlevées, par suite d'une impulsion RESET ou IRQ, ce qui amènerait le condensateur 69 à se décharger trop rapidement.

Le circuit de numéro de série de site 108 fournit un  
numéro de série codé en matériel de 8 bits, spécifique à chaque  
5 installation. Le numéro est codé en coupant une ou plusieurs des  
pistes de numéro de série de site 116. Le micro-ordinateur adapte  
le numéro de série de site matériel à 8 bits avec 8 des 16 bits  
dans le numéro de série de site logiciel sur la carte d'initialisa-  
tion, empêchant ainsi une carte d'initialisation d'une  
10 installation d'être utilisée ailleurs (il y a seulement 1 chance  
sur 254 pour que cela marche - puisque les numéros série de site 0  
et 255 sont ignorés - et permette à une carte d'option sans piste  
coupée de s'adapter à toute carte d'initialisation si on le  
souhaite).

15 Le numéro de série de site est lu en appliquant  
l'alimentation  $V_5^+$  à un circuit multiplexeur 117, le conducteur de  
sélection PA 7 étant à bas niveau. Les données sont alors lues sur  
les conducteurs PB 0 à PB 7.

L'interface série 109 sert d'interface entre le  
20 micro-ordinateur 51 et un terminal portable, tel que le terminal  
de la société NEC numéro 8201 A. Le terminal portable est utilisé  
pour télécharger des informations de suivi de vérification à partir  
de la puce RAM d'horloge/calendrier (telles que la date et l'heure  
des derniers essais de carte (avec succès ou non) pour accéder à  
25 la serrure) et pour régler l'horloge dans la puce RAM horloge/ca-  
lendrier directement, au lieu de le faire par l'intermédiaire  
d'une carte de programmation coupée au niveau de la console. Le  
conducteur CLK1 fournit une horloge synchrone pour des données  
d'émission (sur le conducteur TXD1) et reçoit des données  
30 (conducteur RXD1). Les transistors 118 et 119 fournissent un  
courant suffisant pour commander les conducteurs de sortie.

On a décrit des modes de réalisation particuliers du  
système de verrouillage électronique selon la présente invention  
comprenant la caractéristique spécifique de séparation entre les  
25 fonctions de sécurité et de message de données ainsi qu'un système  
de cryptographie à clef publique et un protocole flexible qui sont  
utilisés pour actionner le système de verrouillage. L'homme de

2597142

40

l'art pourra facilement en déduire diverses variantes qui restent  
dans le domaine de l'invention.

REVENDEICATIONS

1. Procédé pour effectuer sélectivement l'actionnement d'une serrure électronique commandée par ordinateur, caractérisé par la validation d'un message de données codées dans un milieu de mémorisation portable présenté à la serrure, et caractérisé en outre par les étapes suivantes :

- (a) appliquer une clef de cryptographie privée pour coder le message de données ;
- (b) mémoriser le message de données codées dans le milieu de mémorisation portable ;
- (c) utiliser l'ordinateur de la serrure, en appliquant une clef de cryptographie publique pour décoder le message de données codées et déterminer son authenticité ; et
- (d) si le message est authentiqué, actionner la serrure en conformité avec le message de données mémorisées.

2. Procédé selon la revendication 1, comprenant en outre l'actionnement de la serrure sur la base d'un milieu fourni séquentiellement indépendamment du défaut d'utilisation d'un autre milieu fourni précédemment dans la séquence, caractérisé en ce qu'il comprend les étapes suivantes :

- fournir à la serrure un numéro de séquence ( $S_L$ ) ;
- fournir au milieu un numéro de séquence ( $S_C$ ) ;
- comparer  $S_L$  à  $S_C$  ; et
- si  $S_C = S_L$  ouvrir la serrure.

3. Procédé selon la revendication 1, comprenant en outre l'opération de mise en oeuvre de la serrure sur la base d'un milieu fourni séquentiellement indépendamment du défaut d'utilisation de tout milieu fourni précédemment dans la séquence, caractérisé en ce qu'il comprend les étapes suivantes :

- mémoriser un nombre de pont (b) dans la serrure ;
- munir la serrure d'un numéro de séquence ( $S_L$ ) ;
- munir le milieu du numéro de séquence ( $S_C$ ) ;
- comparer ( $S_L$ ) à ( $S_C$ ) ;
- si  $0 \leq (S_C - S_L) < b$ , ouvrir la serrure ; et

si  $0 < (S_C - S_L) < (b)$  mettre à jour  $S_L$  à  $S_C$ .

4. Procédé selon la revendication 1, comprenant en outre l'opération de mise en oeuvre de la serrure sur la base d'un milieu fourni séquentiellement, indépendamment du défaut d'utilisation de tout milieu fourni antérieurement dans la séquence,

caractérisé en ce qu'il comprend les étapes suivantes :  
fournir un numéro de pont négatif ( $b_n$ ) dans la serrure ;

munir la serrure d'un numéro de séquence ( $S_L$ ) ;  
munir le milieu du numéro de séquence ( $S_C$ ) ;  
comparer  $S_L$  à  $S_C$  ; et  
si  $S_C$  est inférieur à  $S_L$  d'une différence non supérieure à  $b_n$ , ouvrir la serrure.

5. Procédé selon la revendication 4, caractérisé en ce qu'il comprend en outre l'étape consistant à mettre à jour  $S_L$  à  $S_C$  si  $S_C$  est supérieur à  $S_L$ .

6. Procédé selon l'une quelconque des revendications 1 à 5, caractérisé en ce que la clef publique est  $n$  et est le produit de la clef privée, deux entiers premiers ( $pq$ ) ; le message de données est  $m$  ; le message codé est  $x$ , choisi de sorte que ( $x' \bmod n = m$ ) ; et l'étape de décodage du message de données implique la réalisation de la fonction  $x' \bmod n$ .

7. Système de serrure adapté à fonctionner sur la base du codage et de la vérification d'un message de données porté par un milieu d'enregistrement discret telle qu'une carte magnétique (32), caractérisé en ce qu'il comprend :

un premier moyen d'ordinateur adapté à appliquer une clef cryptographique privée pour coder le message de données ;

des moyens (27) pour écrire le message de données codées sur le milieu ;

des moyens de serrure (30) comprenant un pêne (33), cette serrure agissant en réponse à la vérification du message de données codées pour ouvrir le pêne ; et

un second moyen d'ordinateur dans la serrure pour appliquer une clef de cryptographie publique au message de données codées pour vérifier le message de données.

8. Système de serrure selon la revendication 7, caractérisé en ce que la clef publique est  $n$  et est le produit de la clef privée, deux entiers premiers  $p$  et  $q$  ; le message de données est  $m$  ; le message codé est  $x$  choisi de sorte que  $x^2 \bmod n = m$  ;  
5 et la vérification du message de données codées est obtenue à partir de  $x^2 \bmod n$ .

9. Serrure électronique selon l'une des revendications 7 ou 8, caractérisée en ce qu'elle comprend en outre :

(a) un moyen d'actionnement pour faire sortir ou  
10 rétracter le pêne (33) ;

(b) un moyen de solénoïde (78) pour relier sélectivement, le verrou au moyen d'actionnement ;

(c) un premier condensateur (62) pour fournir du  
15 courant au moyen de solénoïde pour actionner ce moyen de solénoïde ;

(d) le second moyen d'ordinateur comprenant un moyen de microprocesseur (51) adapté à commander l'application du courant sur le solénoïde pour connecter et déconnecter sélectivement le pêne au moyen d'actionnement ;

(e) un premier bus d'alimentation ( $VM^+$ ) pour fournir  
20 une alimentation au microprocesseur et comprenant un second condensateur (69) pour alimenter le microprocesseur dans le cas de défaut de fonctionnement du premier bus d'alimentation ; et

(f) un second bus d'alimentation (VBATT) adapté à  
25 alimenter le premier condensateur.

10. Serrure électronique selon la revendication 9, caractérisée en ce que le microprocesseur comprend une sortie pour fournir une première impulsion à un instant prédéterminé pendant ou à la fin d'un cycle de fonctionnement, et en ce que la serrure  
30 comprend en outre :

(g) un premier moyen monostable (66) actionnable pour appliquer une seconde impulsion au microprocesseur pour remettre à zéro le microprocesseur à un état actif à partir d'un état de repos ;

35 (h) un second moyen monostable (67) connecté entre le premier moyen monostable dans le microprocesseur et actionnable

par la seconde impulsion pour appliquer une troisième impulsion au premier moyen monostable pour invalider le premier moyen monostable pendant la durée, ce second moyen monostable étant connecté à la sortie du microprocesseur pour être validé par la

5 première impulsion pour revalider le premier moyen monostable ;

(i) un troisième bus d'alimentation ( $VW^+$ ) ; et

(j) un commutateur (71) pour relier sélectivement le troisième bus d'alimentation au premier moyen monostable pour actionner le premier moyen monostable et appliquer ladite seconde

10 impulsion.

11. Serrure électronique selon la revendication 10, caractérisée en ce qu'elle comprend en outre :

(k) un moyen comparateur (72) ayant une sortie connectée au microprocesseur et ayant une entrée de non-inversion connectée à une première tension de référence (87) ;

15

(l) un moyen diviseur de tension (89) connecté entre le premier condensateur (62) et une entrée d'inversion du moyen comparateur pour fournir une seconde tension sensiblement égale à la première tension quand une tension du second bus d'alimentation d'un niveau minimum prédéterminé est appliquée au premier condensa-

20 teur pour produire un signal de sortie de comparateur vers le microprocesseur de façon indicative du niveau de tension du second bus d'alimentation.

12. Serrure électronique selon la revendication 11, caractérisée en ce qu'elle comprend en outre :

25

(m) une résistance (64), le premier condensateur (62) et cette résistance étant connectés au second bus d'alimentation pour fournir une constante de temps (RC) ; et

(n) des moyens pour détecter quand la sortie du comparateur dépasse un nombre prédéterminé de constantes de temps (RC).

30

**This Page is Inserted by IFW Indexing and Scanning  
Operations and is not part of the Official Record**

## **BEST AVAILABLE IMAGES**

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

☐ **BLACK BORDERS**

☐ **IMAGE CUT OFF AT TOP, BOTTOM OR SIDES**

☒ **FADED TEXT OR DRAWING**

☐ **BLURRED OR ILLEGIBLE TEXT OR DRAWING**

☐ **SKEWED/SLANTED IMAGES**

☐ **COLOR OR BLACK AND WHITE PHOTOGRAPHS**

☐ **GRAY SCALE DOCUMENTS**

☐ **LINE(S) OR MARKS ON ORIGINAL DOCUMENT**

☐ **REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY**

☐ **OTHER: \_\_\_\_\_**

**IMAGES ARE BEST AVAILABLE COPY.**

**As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.**